U.S. DEPARTMENT OF COMMERCE
National Technical Information Service

PB-271 925

VIEWING KNOWLEDGE AS A RESOURCE IN FEDERAL DEPARTMENTS OF THE U.S. GOVERNMENT

JAMES F. BERRY, ET AL

U.S. DEPARTMENT OF AGRICULTURE
WASHINGTON, D.C.

SEPTEMBER 1977

PB 271 925

VIEWING KNOWLEDGE AS A RESOURCE IN FEDERAL DEPARTMENTS OF THE U.S. GOVERNMENT

James F. Berry and Craig M. Cook

Economic Research Service

U.S. Department of Agriculture

September 1977

Additional copies available from—
National Technical Information Service
See inside front cover for details.

BIBLIOGRAPHIC DATA SHEET	1. Report No. AGERS-41	2.	3. Recipient's Accession No. PB-271 925
4. Title and Subtitle VIEWING KNOWLEDGE AS A RESOURCE IN FEDERAL DEPARTMENTS OF THE U.S. GOVERNMENT			5. Report Date September 1977
			6.
7. Author(s) James F. Berry and Craig M. Cook			8. Performing Organization Rept. No. AGERS-41
9. Performing Organization Name and Address Economic Research Service U.S. Department of Agriculture 500 12th Street Washington, D.C. 20250			10. Project/Task/Work Unit No.
			11. Contract/Grant No.
12. Sponsoring Organization Name and Address			13. Type of Report & Period Covered Final
			14.

15. Supplementary Notes

16. Abstracts

Knowledge (as opposed to data or information) is proposed as a basic resource of Departments of the Federal Government. Such a valuable resource needs to be viewed on an organization-wide basis. A theory of knowledge is presented as a foundation for many of the concepts involved with establishing a knowledge resource for a department. This theory provides a framework and a vocabulary for the subsequent discussion on implementation. Various types of technology which might be useful in building and using a knowledge resource are explored. Finally a model of a people/machine system which can be used to support a knowledge resource is proposed. In an appendix, potential problems in the construction and use of a knowledge resource are listed, and several areas where further research and development are needed are discussed.

17. Key Words and Document Analysis. 17a. Descriptors

Systems models
Systems management
Information
Information retrieval
Information management
Man machine systems
Information systems
Information theory
Government

17b. Identifiers/Open-Ended Terms

Knowledge
Knowledge management
Knowledge-based systems
Information management
Data base management
Government data processing

PRICE
Papercopy
Microfiche

17c. COSATI Field Group 05-A, 05-B, 05-H.

18. Availability Statement	19. Security Class (This Report)	21. No. of Pages
This publication available only from: National Technical Information Service, 5285 Port Royal Road, Springfield, Virginia 22162	UNCLASSIFIED	
	20. Security Class (This Page) UNCLASSIFIED	22. Price PCA07 - A01

VIEWING KNOWLEDGE AS A RESOURCE IN FEDERAL DEPARTMENTS OF THE U.S. GOVERNMENT

James F. Berry

U.S. Department of Agriculture

Craig M. Cook

University of Maryland

September, 1977

ABSTRACT

Knowledge (as opposed to data or information) is proposed as a basic resource of departments of the Federal governement. Such a valuable resource needs to be viewed on an organization-wide basis. A theory of knowledge is presented as a foundation for many of the concepts involved with establishing a knowledge resource for a department. This theory provides a framework and a vocabulary for the subsequent discussion on implementation. Various types of technology which might be useful in building and using a knowledge resource are discussed. The organizational implications of a knowledge resource are explored. Finally, a model of a people/machine system which can be used to support a knowledge resource is proposed. In an appendix, potential problems in the construction and use of a knowledge resource are listed, and several areas where further research and development are needed are discussed.

Key Words and Phrases: knowledge, knowledge management, knowledge-based systems, information management, data base management, government data processing.

CR Categories: 3.53, 3.60, 3.70, 4.33

Preceding page blank

"Knowledge will forever govern ignorance and the people who mean to be their own governors must arm themselves with the power which knowledge gives."

James Madison

"Knowledge is Power."

Francis Bacon

"Philosophy means to search for clearness where common people do not suspect that there is any lack of it."

William James

TABLE OF CONTENTS

TABLE OF CONTENTS

1. INTRODUCTION

Each day at the various Federal departments hundreds of people ask or are asked questions similar to the following:

* Are certain types of crime related to unemployment figures?

* What agencies are managing programs relating to income transfer?

* Will the data obtained from the LANDSAT satellite program justify the cost?

* Do increases in the the school lunch program lead to increased achievement by the students who participate in the program?

* How much of the oil consumed in the United States is produced from domestic sources?

The ability to answer questions such as these determines, to a large degree, the success of a governmental department. Yet, the answers do not come easily. Part of the reason lies in the type of problems which the public has begun to expect its government to solve. In a world where societies have become increasingly interdependent, the solutions to problems have become progressively more difficult. Yet, the government has found that it cannot afford to increase without bound the number of people, the amount of dollars, and the quantity of equipment that it employs to solve society's problems or even to obtain the answers to questions such as those listed above. In personnel, for example, comparing the number of people in the Federal work force to the number of people in the overall work force produces figures which have remained relatively stable over the past 30 years. The percentage of Federal workers in the total work force was 3% in 1947, 4% in 1964, 3% in 1974 and 3% in 1976 [1]. Consequently, departments have not been getting real increases in the numbers of people which are sufficient to offset the growing volume of work resulting from increased responsibilities. One manifestation of this is an increasing workload which has resulted in rapidly growing amounts of data that must be processed and turned into information for the department's consumers. In producing information products, a department employs a vast storehouse of disparate knowledge concerning factual events, analytic techniques, and organizational goals. An understanding of and improvement in this process is the subject of this paper.

Traditionally the governmental departments have turned to electronic computers for help in attempting to cope with their flood of work, and the computer has responded by demonstrating a prodigious capacity for processing data. That this data processing has been a "help" is, perhaps, debatable when one considers

the massive increase in the number of government forms which are currently needed to feed this new "assistant." The countless side feet of paper output, the growing number of arbitrary retrieval languages, and the almost total incompatibility of data on different machines are other manifestations of some of the difficulties which have come along with the widespread introduction of this technology. Moreover, orders of magnitude reductions in the cost of computing and storage have accelerated the pace of the introduction of electronic data processing by allowing the government to purchase more computing power and storage for the same dollars. Yet with each acquisition the question remains the same: "Is the government really increasing the productivity of its departments by such acquisitions?" Will the people in the Department of Transportation, for example, have more skill, expertise, or knowledge in transportation systems because of their use of computers? The ability to collect, massage, and store data has greatly improved over the last decade, but many departments seem to be acquiring far more data than they can intelligently use. The logistics of keeping track of what data they have on which topics is overpowering and threatens to get worse as more and bigger systems are installed at the various departments. Computer networks (e.g., ARPANET, INFONET, EFTS, the defunct FEDNET) are now possible which can link together many large-scale systems. Keeping track of which systems do what and what data is stored where will become a major technological issue in the very near future. The purpose of this paper is twofold: (1) to address what can be done to enable a department to keep up with an ever-increasing volume of data, and (2) to propose a future which emphasizes providing new generations of tools to improve the quantity and quality of information which a department can produce. This paper will show how computer technology can be used to formalize, analyze, extend, and preserve many of the skills, expertise, and knowledge which are so important to any Federal Department.

Organizations in industry and government have begun to recognize some of the symptoms of their computer-related problems. Some of these organizations have chosen to view data as a corporate resource [2,3,4,5,6,7]. An outgrowth of these endeavors has been the development of the concept of an "enterprise." For the purposes of this report an enterprise is defined to be any total organization (e.g., a government agency, a large corporation, or a small company) which commonly pools its resources in order to achieve some collective goals. More and more, management in these organizations is beginning to enforce constraints on the subdivisions of their enterprise which, prior to the instigation of the data-as-a-resource philosophy, were thought to "own" the data bases of the organization individually. At the same time, these enterprises have been expending increasing amounts of time and money in an effort to make their data, especially computerized data, generally available to all of their members. In many organizations where data has been recognized as a valuable resource of the enterprise on a par with personnel, money, material, and facilities, the need has surfaced to manage this resource actively and effectively from a global or

enterprise point of view. Needs such as these have led to the recent rapid growth in the establishment of corporate data bases and the use of associated software (data base management systems). Many Federal departments are presently in such a growth stage as is evidenced by the increased use of and interest in data base management systems by a large number of organizations. These needs have also led to an introduction of shared data networks as mentioned above. Network technology is generally becoming accepted as a reasonable way for an enterprise to insure that the valuable data resource will be available equitably to all of its subdivisions. In this paper we suggest that the real resource which a department should be seeking to understand and extend is not just its data but its knowledge. Extending the data-as-a-resource philosophy to the concept of knowledge-as-a-resource can be a complicated process, but one which we feel can yield real benefits to a department. In this paper we introduce this new concept and outline some steps which can be taken to use this resource effectively.

Acceptance of the concept of treating knowledge as a resource, of course, will not be automatic. Organizations are loath to change the manner in which they operate, and it generally requires a confrontation with several major problems before such a philosophy will even be considered. We suggest that the following set of problems have either reached a critical state in many departments or will soon become pressing enough to command widespread attention and will, therefore, set off a search for solutions. Many solutions may be found in the philosophy of treating knowledge as a resource.

1.1 PROBLEMS

Several crucial problems are appearing as the government moves from the world of stand-alone data processing to the world of interconnected computer networks. These problems include increasing costs, individual privacy, lack of security, freedom of information requirements, data sharing, multiple user interfaces, inadequate problem-solving support, and the loss of governmental expertise. These problems have been compounded by the lack of a unifying philosophy which could have been used as the basis for architectural planning and systems development. These problems are just mentioned briefly here. They will be discussed in greater detail at the appropriate time in the paper.

1.1.1 COSTS

As indicated in the preceding section, the trend in the various departments over the past few years has seen increasing pressure to produce more and better information with the same or fewer (but more expensive) people. During this period, the cost of computer storage and hardware has continued on a steady decline so that departments are able to buy more and more computer power with constant dollars. In many instances, this increased computer power is being distributed closer to the individual government workers and away from large centralized

computer complexes. There has been a very real shift in total systems cost away from the hardware and toward the software and user-time spent in accessing the computer systems. The problem now is how can an organization place quantitative values on the cost of humans versus machines and how can they evaluate the trade-offs intelligently. Furthermore, the introduction of computer networks will make available to users a large amount of (cheap) computing power and a huge amount of data. The organization and effective management of this dispersed computing power will, itself, present significant problems.

1.1.2 PRIVACY

Privacy is an issue which has received considerable attention in recent years and will continue to grow in importance in the years ahead. Dr. Willis Ware of the Rand Corporation described the basic privacy issues thusly: "An individual has given personal information to a record-keeping system in exchange for some right, privilege, benefit, or opportunity... The individual, in his action to provide personal information, does so with the expectation that it will be used for the purpose for which he gave it. It is his implicit expectation that such information will be used in his best interest and certainly not in any way to his detriment. He does not expect to be annoyed, pressured, harassed, or harmed by its use." [8]

An individual's right to privacy was brought to the attention of government departments with the passage of the Privacy Act of 1974 [9]. This act instructs departments to maintain their record-keeping systems in a manner which will result in accuracy, relevance, timeliness, and completeness. It specifies that individuals may bring civil actions against Federal agencies for any damages which might occur as a result of actions which violate an individual's rights which are protected under the law. It also provides for personal penalties for officers and employees who willfully reveal material whose disclosure was prohibited by the act. Since almost all governmental record-keeping systems are covered by this act, departments are discovering that they need to construct their record-keeping systems in a manner which meets the requirements of the law.

1.1.3 SECURITY/INTEGRITY

Dr. Ware [8] has defined security as: "(a) the protection of a computer system per se and the information within it against accidental or deliberate damage, (b) providing such information only to authorized users, and (c) doing so on a timely basis." Security has been an issue of concern to some government people for a long time, especially those who handle classified information. However, the passage of the Privacy Act of 1974 mandated that all the various agencies begin to consider the security of their record-keeping systems to guarantee the privacy of the individuals involved. The act states that "Each agency ... shall ... establish appropriate administrative, technical, and physical safeguards to insure the security and confidentiality of records

and to protect against any anticipated threats or hazards to their security or integrity which would result in substantial harm, embarrassment, inconvenience, or unfairness to any individual on whom information is maintained" [9]. These security requirements are much stronger than those which may have been required in the past. Satisfying these requirements will necessitate substantial technical improvements in the way (computerized) data is guarded.

1.1.4 FREEDOM OF INFORMATION

The objective of the Freedom of Information Act (FOI) of 1966 is to encourage and mandate the sharing of data, information, and knowledge between the government and its citizens. This act gives all persons a judicially enforceable right to see all records of Federal agencies except to the extent that the records may be covered by an "exemption." The requester need give no reason for the request. Thus, some of the objectives of the FOI may be in conflict with the objectives put forth in the Privacy Act. In seeking to sort out how these conflicts are to be resolved, a Code of Fair Information Practices [10] has been drawn up. The Code states that:

* Protection should be limited to data identifiable with or traceable to specific individuals.

* Protection should be specific enough to qualify for nondisclosure exemption under the Freedom of Information Act.

* Protection should be available for data in the custody of all statistical reporting and research systems whether supported by Federal funds or not.

* Federal law should be controlling; no state statute should interfere with the protection provided.

* Either the custodian or the individual about whom the data is sought should be able to invoke the protection, but only the individual should be able to waive it.

The result is that Federal Departments are faced with rather difficult decisions concerning the data, information, and knowledge which they might wish to place into their record-keeping systems.

1.1.5 DATA SHARING

The problems of access and sharing among various computers with their individual data representations and data storage techniques cannot be solved by technology alone. Privacy, security, integrity, and freedom of information rights also must be considered. In addition, a deeper understanding of the structure and meaning of records and data bases is required before stored data can be shared effectively by many different individuals.

However, the autonomous nature of the various government disciplines guarantees that individual pockets of data will remain almost unintelligible to persons who are not steeped in the lore of the discipline of the government analysts who "own" the data. There is a need to develop mechanisms which will enable people to use the government data (which is available under the FOI). These mechanisms should require minimal amounts of cross-training in multiple job skills. In addition, the difficulties of knowing what data is available and then of locating and using it in a computer network are already well-known to network users and will become widely known as computer networks spread. The problem will become even more acute when citizens attempt to obtain data from the government. We believe that the data access and sharing problems just now surfacing will become even more significant as the government places more reliance on computerized systems in the years ahead.

1.1.6 MULTIPLICITY OF USER INTERFACES

The problem of a multiplicity of interface types (e.g., programs, retrieval languages, or forms) and a multiplicity of kinds within each type (e.g., multiple retrieval languages) has already surfaced among computer populations (witness the ARPANET or the collection of terminals which confront the users of INFONET, for example). Such multiplicity, while useful, requires extensive training of government employees and private citizens in areas extraneous to their primary disciplines. In addition, these interfaces are often unnatural to their human users and they provide a limited view of the data (i.e., a view forced upon the human by the computer which is being used to manage the data). A more human-oriented approach to user interfaces is needed.

1.1.7 INADEQUATE PROBLEM-SOLVING SUPPORT

Today's computerized analytical tools are frustratingly inflexible and provide almost no support for the less algorithmic problem-solving activities of government personnel (i.e., analysis). Statistical subroutines, for example, often are useful only for a specific type or format of data. Documentation on their use is all too often inadequate. Such tools in their present form are almost impossible to share and result in costly duplication of effort. More flexible tools which support analytical processes are required.

1.1.8 LOSS OF EXPERTISE

The services of many of the government's most renowned experts in the various disciplines such as health, transportation, or agriculture continually are being lost because of retirement or resignation. Indeed, when those experts leave the Federal service they take with them 10, 20, or perhaps 30 years of precious knowledge obtained through their experiences in particular applications areas. Evidence that the current means of retaining and transferring their expertise is inadequate is manifested as the same mistakes are repeated over and over again

during subsequent attempts to obtain solutions to recurring problems. (This is not to discount the worth of new approaches to old problems by new employees, but rather to supply them with sufficient information about what has been tried before.) An integral part of this problem is the methodology for preserving the unique skills found in the various departments. We presently attempt to preserve skills through publications, professional courses, textbooks, and on-the-job training. Each of these approaches, however, is labor-intensive and not an integral part of a department's mission. Instead, they are generally regarded as overhead functions which will be performed "when someone gets around to it." A more effective way of preserving the technical knowledge which analysts and experts possess is desperately needed.

1.1.9 NO UNIFYING INFORMATION SYSTEM PHILOSOPHY

No unifying philosophy seems to exist which can be used as the basis for an overall information systems architecture to support knowledge production and use. For the most part, analytic disciplines in the various departments are autonomous (a result, perhaps, of intensive professionalization programs) and this is reflected in the difficulty that arises when trying to communicate across disciplines. In addition, projects and programs are themselves largely autonomous with not much attention being paid to other projects in designing new systems (a result, perhaps, of project management orientation). Finally, there is the continuing problem of imprecise requirements which reflect the natural uncertainty which is indigenous to the types of problems under the purview of government. In too many cases, by the time a computerized information system has been developed to support a given area or program, it no longer meets the requirement because the circumstances for its use or the laws dictating its use have changed. Unfortunately, computerized systems are many times too inflexible to accommodate the necessary changes and continue to survive, forcing humans to change to meet the system's demands. The problem is compounded when such systems interface with the public.

1.2 KNOWLEDGE RESOURCE SOLUTION

As our nation moves into increasingly more difficult times, the quality, quantity, and availability of knowledge in the public sector will become more important. Yet, we must move cautiously in these areas because tightly held knowledge often has served as a significant source of power for the government bureaucracy, and actions which would tend to result in further increases in bureaucratic power are not desirable. However, such problems are not new and many great thinkers in our nation have put forth solutions. The quote from James Madison which precedes this paper suggests one such solution. We believe that Madison was advocating power sharing through knowledge sharing and we also believe that knowledge sharing can best be achieved through the creation of departmental "knowledge resources." Furthermore, we feel that a "knowledge resource" as we define it below will

promote solutions to many of the problems identified above.

For proof that the issues of knowledge and its use are timely and relevant, one only need turn to the current public administration literature where the relationships between knowledge, society, and the bureaucracy have been examined under the topic of "knowledge management" [11]. In our view, and for the purposes of this paper, the term knowledge management has a more limited and technically-oriented meaning, namely the process whereby knowledge (as defined in Section 2) is acquired, maintained, transferred, and preserved as a basic resource of an enterprise. However, even in this limited context the term knowledge management must not be taken to imply control and manipulation of public knowledge nor should it suggest that we might be proposing that the government increase its power over its citizens by "managing all human knowledge." We do not advocate either position and therefore, we shall concentrate on the proper management of the existing knowledge resource of an enterprise. It is our hope that the creation of an organizational knowledge resource will help to promote widespread knowledge sharing within governmental departments and also between the departments and the citizens whom they serve.

Establishing a knowledge resource will require a set of concepts which are meant to promote the understanding, use, and sharing of the knowledge which is already found in a given Federal department or agency. To be meaningful, the concept of a knowledge resource requires a theory of knowledge (which we present in Section 2) to provide a vocabulary and framework for addressing the subject. To be useful, a knowledge resource requires a model upon which an organization can build people/machine systems to improve the processes of acquiring, handling, using, sharing, and preserving the knowledge which is essential to the operation of that organization. We discuss such a model in Section 8. The basic idea behind a knowledge resource, then, is that the knowledge (including the data and the procedures which are employed to manipulate that data) which is found in government departments is understandable and definable. Furthermore, it is cohesive enough to allow a department to construct systems capable of improving the organization's production, use, and sharing of knowledge. We contend that the explicit recognition of knowledge as an organizational resource can have a large impact on the design and construction of computerized systems which might be used to assist in the processing of an organization's knowledge. However, progress will not come easily and considerable work in many different areas will be required before effective knowledge resources can be built by the various Federal departments. In the sections that follow we attempt to:

(1) develop a foundation for the concepts of a knowledge resource (Sections 2 and 3),

(2) relate how government activities could use a knowledge resource (Section 4),

(3) define the roles of various pieces of existing technology (Sections 5 and 6),

(4) suggest some steps which a department can take to begin to build an effective knowledge resource (Sections 7, 8, and 9), and

(5) indicate particular areas which will require additional research (Appendix 1).

Our set of knowledge resource concepts, we feel, can begin to provide solutions to the problems of cost, privacy, security, integrity, freedom of information needs, data sharing, inadequate user interfaces, inadequate problem-solving support, and loss of expertise. Finally, we expect that the concepts which will be needed to create a knowledge resource will be able to provide the foundation upon which a unifying systems architecture can be developed.

The discussion contained in this report is primarily aimed at the Federal Government. This orientation is due, in large part, to the background of the authors and their familiarity with government problems. The application of the ideas contained herein to areas other than those mentioned in the body of this report will, at least for the present, be left to the reader. For those whose orientation and thinking are more toward the corporate or business side, we recommend an earlier paper entitled "Managing Knowledge as a Corporate Resource" [12] which discusses some of these issues.

2. KNOWLEDGE TAXONOMY

If a department is to conceive of knowledge as an organizational resource, it must have some means of understanding and classifying the various types of knowledge which it employs in fulfilling its departmental role. The totality of an organization's knowledge has many aspects and means different things to different people. Consequently, it is a confusing subject which does not, in most current organizational environments, seem to have sufficient cohesion to allow the organization as a whole to discuss the components of a "knowledge resource." In order to fill the need for a high-level view of knowledge which has a corresponding set of basic terminology to describe that view, we have developed a generalized taxonomy of knowledge which we believe can be used to understand the knowledge which is found in any governmental department. We do not propose that this taxonomy encompasses all forms of human knowledge. Rather, it is limited to the knowledge which is most relevant to the functioning of a government department or agency. At the same time, the taxonomy is not to be construed as being limited only to knowledge which might today (or in the future) be found in computerized systems (although computers will undoubtedly play a large part in storing and manipulating such knowledge).

For our purposes, knowledge includes what a department knows (i.e., information), what it knows how to do (i.e., techniques), and what it knows about why it does what it does (i.e., wisdom). Figure 2-1 shows a classification of these types of knowledge into three areas: Factual Knowledge, Procedural Knowledge, and Judgmental Knowledge. Factual knowledge deals with the facts or data about the world which are of interest to the department. Procedural knowledge concerns the operations or procedures which are applied to other knowledge in order to organize it better or to transform it into new knowledge. Judgmental knowledge involves the morals, values, and ethics, the rules and regulations, and the goals and objectives to which the people in a department subscribe and which influence their behavior. Each person in a department, from clerk to chief, employs all three types of knowledge in conducting his or her daily business. Many times the distinction as to which type of knowledge is being used in a given instance is not clear because:

(1) a fair amount of the knowledge in an organization is hidden (being contained solely within human heads and never written down);

(2) there are hierarchies of knowledge contained within each category (factual, procedural, and judgmental);

(3) each type of knowledge may employ other types as input (for example, a statistical routine needs to have some data as input);

(4) judgmental knowledge, since it involves morals, values, ethics, and the interpretation of rules and regulations, can be somewhat nebulous;

(5) people are not used to sorting out types of knowledge; and

(6) knowledge has different contexts.

This latter point suggests that our classification of knowledge into general forms or types could be be improved by considering the subject matter (or context) to which the knowledge pertains. This expansion is discussed in Section 3.2. The primary advantage of recognizing various contexts is the conceptual separation of subject matter according to the needs and purposes of the individual making use of the knowledge. An understanding that knowledge has different contexts is crucial to the development of computer systems which are meant to assist in the building and the use of a knowledge resource. Developers of computerized information systems in the past apparently have failed to recognize the differences which are pointed out in the taxonomy, and as a result, the systems generally have not been responsive to the knowledge requirements of the users. We contend that consideration of the taxonomy in Figure 2-1 and the contexts in Figure 3-1 can be useful in pointing out the differences among the various types and contexts of knowledge and the need to keep them as separate as possible when developing computer systems (a situation which we term "knowledge independence" - see Section 3.1).

```
                              KNOWLEDGE
                                  |
                                  |
                                  |
                                  |
         +------------------------+------------------------+
         |                        |                        |
         |                        |                        |
         |                        |                        |
         |                        |                        |
      FACTUAL                PROCEDURAL               JUDGMENTAL
         | (What)                 | (How)                  | (Why)
(infor-  |               (tech-   |              (wisdom)  |
 mation) |                niques) |                        |
    +----+----+              +----+-----+             +----+----+
    |         |              |          |             |         |
    |         |              |          |             |         |
    |         |              |          |             |         |
    |         |              |          |             |         |
  DATA    METADATA      ALGORITHMIC  HEURISTIC   CONSTRAINTS   GOALS

(facts)   (object        (data pro-  (problem-   (evaluating) (planning)
          relation-      cessing)    solving)
          ships)
```

KNOWLEDGE TAXONOMY

Figure 2-1

2.1 FACTUAL KNOWLEDGE

Factual knowledge comprises the "what" component of a department's aggregate knowledge. It deals with specific facts ("data") and the relationships ("metadata") which the department chooses to consider important among the various data items. The data, therefore, reflects someone's perception of reality while the metadata provides a frame of reference which can be used to interpret the facts or the data. The presence of both components is necessary if a factual knowledge base is to be widely useful to the organization. The importance of this statement will be made clearer in the following paragraphs.

2.1.1 DATA (FACTS)

"Data" is defined to be recorded symbols of all types. This broad definition of data is meant to include all of the recorded symbols of the organization, including its reports, correspondence, forms, charts, pictures, etc. It also includes the symbols which are recorded on the storage media of the department's computers. These latter items are considered to be part of a department's computerized "data banks."

Most computerized data banks consist mainly of collections of stored values. For example, a data bank may contain instances of line voltages which are associated with various pieces of electrical equipment used by the organization. The stored values of these instances comprise the data. However, the data values, of themselves, are not very useful because they generally are not meaningful except to the people who were responsible for storing them in the first place. (Access to numbers such as "120", "240","6","12", "10", "9.3", etc. is not very meaningful.) Furthermore, it is becoming clear that storing more and more data values whose meaning is known to only a few people is not a satisfactory situation because there also must be some (implied) link between things which exist in the real world and the data values which many individuals have recorded in the data banks of their organization. Merely recording the data value "120", for example, is of little worth unless it relates to some activity in which the organization is interested. Even then, this raw value is very ambiguous to most people in the organization since it is impossible for most of them to tell to what the numbers refer. To overcome this ambiguity, it is useful to associate the value "120" with another object (an attribute) called "volts." An "attribute" is defined to be some property which is common to all of the objects which are in a given class. Today, such association is done mainly in people's heads where it is not accessible to other humans. In any case, the pairing "volts/120", is called an "attribute/value pair." Such pairing is the first step toward reducing the ambiguity of the values in the data base. Ambiguity still exists, though, because the pairing does not indicate which objects in the real-world can have the attribute (property) of "volts" nor which could have the value of "120". However, the creation of an attribute/value pair of "volts/120" has already served to limit the context of the value, "120" to some

measurement of voltage. Thus the attribute/value pair "volts/120" should be related to one or more objects which use voltage (i.e., typewriters, calculators, air conditioners, terminals, computers, etc.). If someone in the organization needs to set up a floor plan which is to specify where the different objects are to be placed, then this person must understand which associations (objects/attributes/values) are correct for each of the objects. If the records on these objects were placed in data banks which contain only values, then access to the data banks will not provide sufficient knowledge about the meaning of the data. The pairing of the objects and attributes with the correct values would have to be provided by a person who understands the data in the data bank. Factual knowledge which is constructed in this fashion ties organizational ability to take an action (drawing up a floor plan) to an ability to contact the person who holds the missing pieces of the needed knowledge. This type of situation can be very harmful to an organization if the right people aren't available at the right times. In the following paragraphs we shall examine how an organization can construct its factual knowledge bases in ways which can minimize unnecessary dependence.

Continuing with the example in the preceding paragraph, "computers" were identified as one of the objects which could have an attribute of "voltage" and someone might have stored "240" in a data bank as the value of the voltage requirement of a particular computer. "Computer," in this case, is said to be a real-world entity about which the enterprise (i.e., the department) desires to record attribute/value pairs of "volts/240." An "entity" is defined to be a person, place, thing, or event that exists in some portion of the real-world which is of interest to the enterprise. In order to define what constitutes a given entity, it is useful to associate certain attributes with each entity. In the previous example, the entity "computer" might be assigned the attributes of "model number", "serial number", "size", "voltage", "line frequency", etc. In data processing terminology the total collection of attributes and values which are associated with a single object might be known as a "record" and each attribute known as a "field." (Unfortunately, the data processing meanings for "records" and "fields" are not restricted to objects and attributes since "records" and "fields" also may consist of arbitrary mixtures of different objects, attributes, and values.) An enterprise which has chosen to organize its factual knowledge along the lines of entities, attributes, and values has taken a first step toward creating a foundation upon which the factual portion of a knowledge resource can be implemented.

There are important conceptual differences between conventional data processing notions (which form the basis for most of today's data banks) and the ideas of entities, attributes, and values. These differences become crucial if an organization decides to begin to build a knowledge resource. They also become important if one believes that one of the main functions of a knowledge resource is to facilitate knowledge sharing.

Organizational adoption of the concept of entities, attributes, and values allows the organization to define data about data, i.e., metadata (see 2.1.2 below). Metadata can be used to form conceptual models which can represent the real-world which is relevant to the organization. This enables humans to work with the conceptual models which are supplied by the metadata portion of factual knowledge without needing to get involved in the voluminous number of instances which might occur in a data bank. The provision for high-level access to conceptual models also plays a major role in more advanced problem-solving activities (see Section 2.2.2) since the metadata can be useful in directing more in-depth searches, i.e., in pruning irrelevant data. For example, providing metadata on the entity "food production" allows an analyst to begin to understand this entity in an abstract way without having to examine the entire data bank. Frequently, many questions can be answered at this high level (and can, in fact, be handled more efficiently). Thus, answers to a query such as "Do we have any facts on food production in Asia?" can be obtained by accessing the metadata bases alone without accessing the actual (facts) data banks. Of course, obtaining specific factual data such as "How much rice was produced in Indonesia in 1975?" would necessitate accessing the factual data banks. (Even so, the metadata can be used to select only those data banks which are likely to contain relevant facts.)

2.1.2 METADATA (RELATIONSHIPS)

A useful capability to have when dealing with entities is an ability to group objects together into a set and to refer to this grouping in a named aggregate. For example, when entities of the same object class have some of the same properties, then they can be grouped into an aggregate called an "entity set" (sometimes called "files" in data processing terms). The collection of all computers owned by the organization is one example of an entity set. The collection of all computers leased by the organization is another entity set. The collection of all departmental equipment is the "equipment inventory" entity set, and so forth. The ability to define hierarchies of entities through the set concept is very powerful and will allow an organization to build complex entity sets which reflect the complexities of the real world.

Quite often it is desirable for an enterprise to be able to define relationships between different entities or sets of entities which reflect their natural association in the real-world (at least the natural associations as viewed by the enterprise). These relationships can provide a frame of reference which could be used to supply meaning to the data. In the example mentioned above, it may be useful for a department to identify a relationship between food production and exports and imports of food. Furthermore, it might be useful to specify a relationship between food production and the strains of plants used in the production. There might be other relationships between the areas where food is produced and the plants which are grown. Relationships can begin to get more obscure, for example, when attempts are made to

relate oil imports and food production (oil can be converted into fertilizer). Metadata, then, is defined to be the aggregate of the entity sets and the relationships which exist among the various entities and entity sets. It also describes the structure of an entity set as well as the format of the entities themselves. To a large degree, metadata is what gives a factual knowledge base its potential for meaning (based on the conceptual model which it represents). In the realm of data base management one generally speaks of a combination of data and metadata as a "data base" while a collection of plain facts or data is often termed a "data bank." We find this a useful distinction and we shall attempt to adhere to it throughout the remainder of this paper.

2.1.3 INFORMATION

Factual knowledge, then, is the term used to describe the entities, attributes, values, entity sets, and relationships which are associated with the real-world objects represented in the data base. All of these things will need to be in a data base which is to be part of a knowledge resource. It is far too easy, however, to create a data bank (not a data base) which is useful to only a very limited number of individuals.

The reason that data, alone, is not very useful is because it is almost devoid of meaning except to the person who recorded it. Recording a fact about a strain of rice, for example, may be meaningful only to a particular agronomist. If the facts are to have wide utility to other people besides this agronomist then these people must be provided more knowledge before they will find this fact to be useful. They will need access to the set of relationships which the agronomist believed to be relevant to the rice entity. The degree to which the agronomist is able to provide metadata will determine how widely his factual knowledge can be shared.

Factual knowledge, even if it is constructed in the manner which we suggest, often has a limited life-span and it can lose its utility with the passage of time. (The concept of time with respect to data bases is a complex issue which will be discussed later in paragraph 2.6). For example, mixes of crops in food production might change, exports may turn into imports, strains of plants may be wiped out by disease, etc. These time-related changes need to be noted and the new relationships recorded.

For many years, departments have been processing data and creating data banks. With the recent advent of data base management systems, some of them have begun to process factual knowledge and they have begun to construct metadata and to create data bases. However, the creation and use of metadata is still in its infancy and organizations have much to learn in this area. All of this activity, especially the move toward the use of data base management systems, might be regarded as steps in the right direction. However, departments can still do much to improve the processing of procedural and judgmental knowledge as well.

2.2 PROCEDURAL KNOWLEDGE

Procedural knowledge concerns the "How" component of a department's aggregate knowledge. It includes their operations, procedures, and problem-solving techniques. In effect, procedural knowledge is what the enterprise knows how to do.

In order to classify this type of knowledge we have found it helpful to employ a continuum of knowledge which was developed by Herbert Simon in 1960 [13]. The extremes of Simon's continuum are labeled programmed and nonprogrammed. In Figure 2-2 we show this continuum. We have elected to use the terms "algorithmic" and "heuristic" to refer to Simon's concepts of programmed and nonprogrammed knowledge to avoid confusion in terminology since we wish to use the words "program", "programming", and "programmed" in their conventional computer context.

Algorithmic procedural knowledge is used for repetitive and routine problems which can be solved by detailed step-by-step procedures. Examples include such things as statistical routines, measurement routines, instructions for using conversion tables, and some scientific formulas. Heuristic procedural knowledge, on the other hand, includes the procedures which are used to solve unstructured problems. Techniques for inferencing, heuristic search strategies, and reasoning (see Section 2.2.2) are examples of heuristic procedural knowledge. Generally, it is not too difficult to distinguish between algorithmic and heuristic knowledge, but, as with all continua, items which fall near the middle frequently will have characteristics of both classifications, and the sharp distinctions tend to become blurred. The two categories of procedural knowledge are explained in greater detail below.

Routine and repetitive problems	Novel, unstructured, elusive, and complex problems
Definite procedures	General problem-solving techniques
ALGORITHMIC (Programmed)	HEURISTIC (Nonprogrammed)

ROCEDURAL KNOWLEDGE CONTINUUM

Figure 2-2

2.2.1 ALGORITHMIC PROCEDURAL KNOWLEDGE (Programmed)

Algorithmic procedural knowledge is used whenever problems are stereotyped and routine and are amenable to being solved by the repetitive application of some step-by-step rules or procedures. Use of this knowledge implies a complete and detailed understanding of the problem and its solution. Algorithmic knowledge frequently can be expressed in a computer program where the programmer is able to specify carefully the passing of control from one piece of program code to another. Every instruction to the computer must be understood and carefully placed in the proper sequence. Federal Departments, by and large, have been very successful in computerizing algorithmic knowledge and, to a large degree, most of the computer power in the various departments is devoted to this type of activity. For example, most of the computing capacity found in the Social Security Administration is devoted to preparing checks and maintaining records (algorithmic activities).

Unfortunately, not all of a department's problems can be solved algorithmically. Analysts employ heuristics or "rules-of-thumb" which they have built up over a number of years of experience in their analytic discipline in order to solve certain problems, to determine which algorithms are relevant, or to construct new algorithms as warranted. For example, the process of determining the eligibility for the social security checks mentioned above would generally be classed as a heuristic process.

2.2.2 HEURISTIC PROCEDURAL KNOWLEDGE (Nonprogrammed)

Heuristic procedural knowledge is used to solve problems or to react to situations when it is not possible to prespecify a particular algorithmic solution. Heuristic knowledge is generally specific to skill areas such as economics, medicine, mathematics, etc. but it can also involve general common sense reasoning as well. Specific heuristic knowledge is used to guide solutions or reactions and it is frequently expressed in either goal-driven or data-driven terms. A transportation planner who seeks to obtain the "best" mix for the expenditure of public transportation dollars (the goal) is operating with goal-driven heuristic knowledge. An economist who is running various statistical tests (algorithmic knowledge) on census results (the data) is operating with a data-driven set of heuristics. The results of each test might suggest certain paths to pursue and close other paths from further consideration. In this latter example, the results of the execution are used in a heuristic fashion to determine what is to be done next.

Another common form of heuristic procedural knowledge is inferencing. There are three types of inferencing which are generally recognized: deductive, inductive, and abductive [14]. Deductive inference refers to the process of reasoning that whatever is true of all instances or members of a class must be true of one instance or member. The premise of the deductive argument is said to provide definite evidence for its conclusion. This

point is, at the same time, its strength and its weakness. If a deduction does not have a major premise which is without exception true, then attempts to use this form of logic will not have the expected inevitability.

Inductive inference is a form of reasoning where a body of facts or observations are used to discover rules or generalizations which, more or less, explain the observed phenomena. Stated another way, induction means to generalize from a number of cases of which something is true, and infer that the same thing is probably true of a whole class. This type of inferencing is directly related to reasoning by analogy and it has much in common with statistical probability theory. An example of inductive inference would be for a doctor to examine the records of all of his polio patients, to observe that all had had high fevers in the early stages of the disease, and then to generalize a rule which stated that one of the probable characteristics of polio is the occurrence of a high fever in the patient. If, unfortunately, the doctor were also to observe that all of the polio patients were homeowners, then he or she might conclude that the state of being a homeowner was also a characteristic of having polio. What this points up is the fact that the premise may or may not provide definitive evidence for the conclusion.

Abductive inference is a kind of reasoning where a hypothesis is formed which, if true, would explain some collection of observed facts. For example, an observation that Jane Smith has spots when taken with the rule that all people who have measles also have spots might lead a doctor to hypothesize, via abductive inference, that perhaps Jane has measles. The problem with arriving at a definite conclusion at this point is the fact that other types of disease might also produce spots. Therefore, a necessary step in this form of inference is to attempt to identify additional evidence which could be used to support the hypothesis. A high fever and a coated tongue also might be characteristics which sometimes are present in people who have measles (and many other diseases as well). Positive or negative indications of symptoms, if taken in combination through the process of abductive inference, may lend strong support to a given hypothesis.

As a further illustration of what is meant by heuristic procedural knowledge, consider the following example of a very simple type of problem: solving a jigsaw puzzle which has 1,000 pieces. A brute force algorithmic solution would involve 1,000 factorial trials and the sheer size of this number dictates that this algorithmic approach is not practical. Therefore, most people do not attempt to solve such a problem with a brute force approach. Instead, they might begin to analyze the problem in order to identify any characteristics which might be exploitable for solution. The results of such exploration might reveal that it is possible to construct stable partial solutions to the problem (parts of the puzzle can be assembled and then used later to build bigger parts). In addition, there seem to be two distinct types of data which will be available during the puzzle-solving

activity: the shape of the contours and the designs printed on the surfaces. Therefore, the problem-solver might hypothesize that it would be possible to place all of the pieces of a particular color in one or more piles or to place all of the edge pieces in another pile. These observations and hypotheses are used by the problem-solver to formulate a problem-solving strategy which consists of formulating the problem in a way in which stable subconstructions can be formed and built upon. (This means to choose sections of the puzzle which would appear to be easier to put together.) Another part of the strategy is to characterize the rules of construction along two or more dimensions (color and shape) which are not perfectly correlated but which are each sufficient (generally) to determine the correct solution. As a person proceeds toward the solution, different facts about the puzzle become evident at various points in the solution process. The circumstances may be such that the data is incomplete, subjective, or erroneous (a situation which is quite prevalent in many real-life problems). For example, a dog may have chewed off the corners of some of the pieces or may have destroyed some of the images. Thus, at any point in time, a person's data concerning the problem or its solution may be incomplete. Nevertheless, the objective is to continue toward the solution by using the subconstructions as they are built. These subconstructions can be used as feedback to help the puzzle-solver understand the entire puzzle before it is constructed. This helps speed the solution along. Thus, while it may not be possible to define a total algorithmic solution to puzzle solving, it is relatively easy to define a set of heuristics which are quite useful in such activity.

As illustrated in the above examples, heuristic knowledge is extremely flexible and is capable of dealing with a broad variety of problems and situations. It is the most common form of knowledge used by government analysts but very little, if any, of this knowledge is presently computerized. Instead, it remains the purview of technical specialists who have received extensive training in such problem-solving areas.

2.3 JUDGMENTAL KNOWLEDGE

Judgmental knowledge includes the "why" component of a department's aggregate knowledge. It deals with the wisdom which is used to determine which factual and procedural knowledge is relevant to a given decision. The exercise of judgmental knowledge (especially in the public sector) also includes consideration of moral and legal issues before making a decision. We believe that there are two types of judgmental knowledge: (1) Constraints (which include Values and Regulations) and (2) Goals (which include Objectives).

2.3.1 CONSTRAINTS (VALUES AND REGULATIONS)

Values are a form of knowledge which are obtained from society and experience. Generally, values consist of the written and unwritten guidelines to moral and ethical conduct which are

expected of employees of the enterprise. The Federal Employees' Code of Ethics, the Code of Ethics for the Association for Computing Machinery (ACM), and the Ten Commandments are examples of values that have been written down. Rules of social behavior or conditions for survival are examples of values which are, perhaps, less formalized. Values do not carry the force of law and cannot be used as a legal defense to justify a decision.

Regulations are a form of knowledge similar to values but which are written down in the form of statutes or rules of conduct which carry the force of law and judicial action. The Privacy Act and the Freedom of Information Act are two well-known examples of such statutes. The Civil Service Regulations are examples of government-wide rules which are meant to guide and control employee conduct at all agencies.

The importance of the need to include moral and legal considerations in the decision-making process has been underlined by recent events involving all branches of government which have revealed questionable ethical or legal conduct. Therefore, we must emphasize that a department needs to provide for the application of judgmental knowledge in its use of a knowledge resource. Our logical system design which is discussed in Section 8 provides a place where humans can exercise judgmental knowledge.

2.3.2 GOALS AND OBJECTIVES

Goals are a form of knowledge which defines the basic mission of a given department. They are often given in broad terms. For example, the Economic Research Service (ERS) of the U.S. Department of Agriculture has the broad goal of developing and providing economic information to members of Congress, USDA policy officials, other government agencies, State and local officials, foreign government leaders, farmers, farm organizations, marketing firms, and farm supply companies.

Broad goals are often broken down into more specific goals which are intended to further refine the definition of the organization's mission. Specific goals which relate to the mission of the ERS, for example, include developing economic information on:

* U.S. food and fiber production
* farm and rural adjustments
* agricultural use of natural resources
* consumer interests relating to agriculture
* foreign trade of agriculture products
* USDA international technical assistance

The goals specified above may be further refined at increasing levels of detail in order to clarify the definition of the organization's mission. If properly done, the goals and subgoals should form a tree which can be traced from top to bottom. Otherwise, it will be difficult for a department (or anyone else for

that matter) to understand why it engages in the activities that it does.

Objectives relate to the specific actions or tasks which are to be completed in connection with particular goals. Essentially, objectives should serve as a means to measure the progress toward some goal or set of goals. When used this way, the objectives generated by the organization are frequently spelled out in Management-by-Objective programs or they may be expressed as milestones in project management charts.

A specific objective which is designed to satisfy the goal of making economic information more available might be to provide "Supply and Demand Estimates" on-line in a computer network which would be readily available to a wide variety of government and non-government users. An objective which is meant to satisfy a goal of providing more timely information might involve changing the update cycle of figures on "Agricultural Statistics" from a yearly basis to a quarterly basis.

Goals and objectives can be externally or internally generated. External goals might be provided by Congress when they pass legislation. Laws often define the areas which the legislators expect will be addressed by their legislation and they also may specify performance criteria which are expected to be met. Such external definitions and criteria might be directly translatable into internal organizational goals and objectives.

The area of judgmental knowledge is the most ill-defined and least understood aspect of the knowledge which exists in an organization. Nevertheless, there is a need to support the human exercise of judgmental knowledge as it relates to the knowledge resource of an organization. Judgmental knowledge should be used to justify the collection, use, sharing and retention of the other forms of knowledge. If some set of facts does not relate to the basic mission of the organization, then they should not be collected or retained. Great masses of personal data on individuals are available but this does not mean that a department should decide to try to collect them. Judgmental knowledge (rules, regulations, laws, ethics) should be used to determine whether such an activity should be allowed to take place.

Portions of what we define as judgmental knowledge may never be placed into any computerized system. Nevertheless, the ability to apply all forms of judgmental knowledge is important and failure by an organization to recognize the need for such consideration can lead to the design of systems and, ultimately, to the creation of a knowledge resource which is not adequate to meet organizational needs.

2.4 KNOWLEDGE ABSTRACTION

Knowledge, in general, includes concepts which are abstracted from other knowledge over a relatively long period of time. It is these processes of abstraction which occupy most of an

organization's time and energy. Organizations are seldom interested in understanding or explaining all of the complexities of the phenomena which exist in the real-world. Instead, they desire to understand only that portion which is relevant to them. Luckily, knowledge is amenable to a "top-down" approach and a knowledge resource can be built in this fashion. Most of what is known in science, for example, was constructed in just this fashion. There was a great deal of knowledge on physical and chemical behavior before there were molecular chemistry theories and they, in turn, preceded atomic theory.

The process of abstraction is important to our concepts of a knowledge resource to reduce the amount of understanding which humans will need to possess in order to use the knowledge in the resource. In the example from the preceding paragraph, it is possible to use molecular chemistry theory without being an expert in atomic theory. The necessary abstractions only need provide an approximate, simplified characterization of atomic theory which is at a lower level of knowledge than molecular chemistry which is, itself, lower in a knowledge hierarchy than theories of chemical behavior. The characteristics of hierarchies of abstractions, and hierarchies of knowledge itself, are capable of being exploited when they are organized into a knowledge resource.

Abstractions also have another characteristic which is useful when building a knowledge resource. They are capable of conveying succinctly substantial amounts of information (witness: any formula in mechanical physics) without requiring the explicit enumeration of all possible instances of the data. This characteristic aids understanding and promotes knowledge sharing since it is possible to provide a lot of knowledge with a very small amount of data.

Abstractions are useful in still another sense. They permit an organization to characterize the objects in their data bases in such a way that it is possible to focus on a few characteristics which might be of general interest to the entire organization. For example, it is possible to abstract out the tensile and compressive strengths of many materials (woods, metals, plastics, etc.) without being forced to worry about the chemical properties of these objects. It is generally not necessary, for example, that all units in the organization know how to calculate these abstractions so long as they are done correctly. By pursuing the development of abstractions, whole hierarchies of abstractions of the knowledge which is relevant to broad areas of the organization can be built up over a period of time.

Of course, the conditions under which an abstraction is valid also must be preserved as part of the knowledge about it. For example, a piece of knowledge might be an abstraction which gives the minimum caloric requirements which are needed by various groups of people to sustain life. Another piece of knowledge might be facts on the allergy of given groups of people to milk products. The first set of knowledge would be useful to

government analysts who are trying to identify base levels which can be used to determine if a food program is supplying enough nutrition to its recipients. This abstraction (minimum caloric requirements) might not be valid for those people who cannot properly digest milk products. Therefore, other knowledge must be used to determine the alternative food choices which might be needed to help balance the diet of persons who cannot consume milk products.

2.5 KNOWLEDGE INTERACTION

There is a very close relationship among factual, procedural, and judgmental knowledge and it is not always obvious to which category a given item belongs. In daily processing done by an organization, data may be massaged by different procedures and additional knowledge abstracted out by different elements to satisfy their own purposes. The data may be kept in a single pool which is accessed in parallel by the various elements, or it may be passed sequentially from one element to the next with each division acting autonomously upon it. There is also the need to pass the factual knowledge (i.e., processed or abstracted data) from one division to another. Each division may take incoming data and process it to create new factual knowledge or new procedural knowledge with the results being passed on to the next group. Where value is added at a given step of the process, it should be possible to preserve that added value (new factual or procedural knowledge) for subsequent steps. As data is passed from group to group, each group, in turn, applies its unique procedural knowledge to manipulate the data further and to derive new knowledge from it before passing it on to the next organization. The process repeats itself many times in the department with each organization receiving data, producing other factual knowledge, and abstracting new knowledge.

By a slightly different process, the procedural and judgmental knowledge which a group develops can also be passed along to other groups. This can be in the form of instruction on techniques, comprehensive summaries of the history of a particular problem area, comments on the significance of certain pieces of information, or, perhaps more importantly, as forecasts about some future events. For example, one group's factual knowledge (an economic forecast by the Economic Research Service) can become another group's data (one of many inputs which Congress uses to decide what legislation is needed). In fact, the receiving group (Congress) may be as interested in the techniques (procedural knowledge) and the data (factual knowledge) which the generating group (ERS) used to create the forecasts as they are in the forecasts themselves. In effect, they are seeking some measure of validity and reliability for evaluating the forecasts which they receive. The formalization of ERS's procedural and factual knowledge is necessary if such measures are to be provided.

Because of the need for interaction, no single piece of knowledge is very useful by itself. Facts without procedures to

manipulate them or without goals to satisfy are of little worth. The converse is, of course, equally true. Goals without data to work on or techniques to apply are empty dreams. Hence, large portions of knowledge of different types (factual, procedural, and judgmental) are necessary before useful work can be performed by a department. Indeed, the degree of interaction among the various types of knowledge in an enterprise reflects the complexity of the activity in which the enterprise is engaged.

There is considerable communication of knowledge between the departments of the legislative and executive branches. This communication has been primarily factual knowledge but recently increasing amounts of procedural knowledge have begun to be exchanged as well. Some of this communication occurs during Congressional hearings, but it can also take the form of reports or estimates prepared by a department. A department also might be called upon to defend the methods of analysis which were used to produce the reports or estimates. In the future, knowledge sharing potentially could take the form of on-line requests for information retrieval. Similarly, information service requests should be expected from the public at large. In any case, the possible benefits of improved knowledge communication throughout government and especially between the government and its citizens make it worthwhile to explore the potential for knowledge sharing.

2.6 THE RELATIONSHIP OF TIME TO KNOWLEDGE

Time is a concept which applies to the entire taxonomy of knowledge. What is known by an organization is relative to some point in time, either the past, the present, or the future. A problem concerning the time dimension can arise because of today's data management software which is used to manage portions of the factual knowledge of an organization. In such software, the concepts of time must be supplied by the human. The system treats everything as though it exists in the present. Questions concerning the past must be referred to archival data bases which then provide the context of "the present" for a particular request. As data bases are connected via computer networks (e.g., ARPANET, INFONET), the concept of time context for each data base will become extremely important. For example, questions on the amount of dollars which are devoted to transportation can have several different answers depending on the time context involved, i.e., which budget year is involved. Because of questions such as the one just mentioned the issue of time context for the retrieval of facts is frequently an important one.

Similar problems can arise with other aspects of time as it relates to dealing with factual knowledge. The concept of time intervals, e.g., "yearly budget figures", requires a notation of time in the data to a level of detail commensurate with the kinds of questions anticipated (yearly figures are not sufficient to answer questions which need a month- by-month breakout).

Changes in the structure of a data base (i.e., the metadata) can invalidate requests for information from that data base. Questions which used to be answerable may no longer be so, or questions which are answerable now may not work against old versions of the data base in archival storage. For example, the Department of Transportation is a relatively new department so that requests for facts concerning the expenditure of transportation funds or for facts which trace the creation of a particular transportation policy would not succeed if attempts were made to seek all answers relative to DOT. Other departments and agencies made transportation policy and expended transportation dollars before DOT was created.

Time dependencies apply to procedural knowledge as well as to factual knowledge. Some heuristic knowledge may be transformed into algorithmic knowledge. For example, observations on rates of types of crime may be compared with certain economic indicators. The results of the comparison may lead the analyst to develop a set of formulas which can project increases in certain types of crime based on specific economic indicators. Algorithmic knowledge which is available may no longer apply to a particular problem area (for example, conversion of Federal workers from their own retirement system to the social security system would make all of the old retirement formulas obsolete). It becomes crucial, then, to keep track of the time associated with procedural knowledge and with the factual knowledge it is to process in order to keep them synchronized.

Judgmental knowledge requires an explicit time context also. Laws, rules, and regulations are subject to change with the passage of time. What is permitted or legal today may not be acceptable tomorrow or it may not have been acceptable yesterday. Concepts of ethics, values, and morals also change over time. Hindsight analysis of past decisions requires an understanding of the environment at the time a decision was made. Future analysis requires the extrapolation of several alternative future environments so that desired paths can be selected. Reactive decisions require an understanding of the way things are today and hence need up-to-date knowledge.

As departments try to preserve more and more of their knowledge in computerized systems, the time dimension will become more important. In the future, requests of the knowledge resource will have to specify a time parameter in order for the knowledge to be meaningful.

3. KNOWLEDGE INDEPENDENCE AND CONTEXT

3.1 KNOWLEDGE INDEPENDENCE

Obtaining a high degree of independence among the various forms of knowledge is a primary goal of knowledge management. By knowledge independence we mean the separation of factual knowledge (data or information) from the procedural knowledge which is used to access or employ that information and the judgmental knowledge which is used to direct the organization's activities. Such a separation is crucial to the long-term survivability and utility of a department's knowledge resource. By providing knowledge independence, facts will be more readily usable by multiple procedures (data sharing), and procedures should be more amenable to running with multiple sets of facts (tool flexibility). The separation of algorithmic from heuristic knowledge can provide a rational basis for understanding the kinds of procedures which must be translated into software for the computer systems. Indeed, the separation of data from procedures and procedures from each other is a basic tenet of the discipline of software engineering [16,17]. Finally, the recognition of the importance of the human role in managing judgmental knowledge is crucial when attempts are made to automate procedures which rely heavily on this type of knowledge. Ultimately achieving knowledge independence, however, will require some canonical self-describing representation of knowledge (not only data, but procedures). This is a research area which is now under investigation at a variety of research institutions [18]. Indeed, many of these other points are being pursued by the industry today (although not under the collective name of knowledge independence). Basic software engineering techniques, data and procedure independence of data base management systems, and procedural knowledge independence of knowledge-based systems all are steps in the direction of general knowledge independence.

Another important aspect of knowledge independence is the need to separate the various subject matter (or context-knowledge) required of analysts so that they can concentrate on their respective areas of specialization (e.g., economics) without wasting valuable time learning extraneous details from other contexts (e.g., computer protocols). We agree that some degree of understanding about other contexts is both useful and desirable for each analyst in a department, but we contend that requiring too much understanding can be counter-productive. The challenge of knowledge management is to discover and develop the right blend of knowledge independence to produce the best possible knowledge environment.

3.2 KNOWLEDGE CONTEXT

In order to make the knowledge taxonomy relevant to a particular enterprise, it is necessary to consider the environment of that organization. By environment, we mean the context or subject matter of the knowledge which is relevant to the enterprise. For the purposes of this paper we find it convenient to divide a

department's corporate knowledge into three contexts: mission, tools and support, and direction. Each of these three contexts can be further divided into knowledge of the appropriate skills and knowledge of a particular target. These areas, in turn, consist of factual, procedural, and judgmental knowledge which are relevant to the particular context. Figure 3-1 depicts this organization of a department's knowledge by context.

```
                          DEPARTMENTAL KNOWLEDGE
                                    |
                                    |
        +---------------------------+---------------------------+
        |                           |                           |
        |                           |                           |
     MISSION              TOOLS AND SUPPORT                DIRECTION
        |                           |                           |
  (applications)            (computers, etc.)            (management)
        |                           |                           |
   +----+----+              +-------+-------+              +----+----+
   |         |              |               |              |         |
   |         |              |               |              |         |
SKILLS    TARGET         SKILLS          TARGET         SKILLS    TARGET

(Economics, grain  (Programming, (Mainframes, (Budgeting, (FY 77 budge
 Medicine,  crops,  Maintenance,  Terminals,   Personnel,  Director's
  ...)      cancer     ...)          ...)         ...)      staff,
              ...)                                             ...)
```

KNOWLEDGE CONTEXTS

Figure 3-1

3.2.1 MISSION CONTEXT

Knowledge in the context of a department's mission involves numerous skills such as economics or statistics. Knowledge also involves understanding a variety of target areas (including geopolitical units such as South America or interest areas such as cereal grain production). Geopolitical units are sometimes used to divide interest areas into smaller sets. The converse is also true. The primary activity contained in the mission context is what is usually termed applications. Analysts (e.g., economists) who are concerned with particular applications (forecasts on cereal grain production) require substantial knowledge about their application. Additional knowledge about the computers used in the tool context can be helpful, but the valuable time spent in training the analyst to function effectively in the computer context, could, perhaps, be better used to improve the analyst's skill or target knowledge. Oftentimes the trade-off is such that the savings in computational time fully warrants the training time required before an analyst can master the machine; but many times it does not. It is important to keep clear the distinction between that portion of the analyst's time which is spent acquiring specialized skills and target knowledge and the time which is spent wrestling with awkward computer protocols and query languages (the tools and support context). The computer can be a great time-saver, but it can also be a great time-consumer. Figures 3-2 and 3-3 show some examples of the different types of knowledge required by skills and target areas in the mission context.

	STATISTICS	ECONOMICS	ENGINEERING	...
FACTUAL	Normal table of distribution	Definition of Gross National Product	Definition of thermal conductivity	
PROCEDURAL	Steps for testing the Null Hypothesis	Method for calculating GNP	Calculation of thermal conductance	
JUDGMENTAL	Samples will be distributed normally	GNP does not measure changes in quality	Empirical coefficients to be considered	

MISSION/SKILLS CONTEXT

Figure 3-2

	MEDICINE	MACRO ECONOMICS	CONSTRUCTION	...
FACTUAL	Nr.of patients with heart trouble	Endogenous variables in U.S. economy	Table of Conductivity and Resistivity	
PROCEDURAL	Procedure for pooling patient data samples	Klein-Goldberger Dynamic Econometric Model	Calculation of insulation payback interval	
JUDGMENTAL	One-tailed stat test will give best results	K-G model will forecast national income	Consumer acceptance of insulation type	

MISSION/TARGET CONTEXT

Figure 3-3

3.2.2 TOOLS AND SUPPORT CONTEXT

The context of tools and support involves skills and target areas which are very different from those needed in the applications relevant to the mission context. In this context we group activities which deal primarily with the development and maintenance of tools. Also included are functions which are done in support of but not directly related to a department's mission. Such activities might include contracting, personnel, desks and furnishings, mathematics, and computers. This last category is of particular relevance to the concepts of this paper, and we shall concentrate on the computer as the primary tool of interest. Skills such as programming, systems, operations, and so forth require considerable training and involve general aspects which apply to a variety of target machines. Targets such as particular mainframes or individual terminal types form the basis of reality to which the general computer skills must be applied. Both kinds of knowledge are important in understanding the computers in the tools and support context. Section 6.3 provides a Physical View of a hypothetical knowledge resource by describing the physical environment of the tools and support context. Such a view is, of course, essential to the overall management of a knowledge resource. Figures 3-4 and 3-5 show some examples of the different knowledge skills and targets required in order to understand fully the Physical View.

From the mission or direction contexts the tools and support context is oftentimes viewed as an overhead function, and the users in these areas generally resent being required to master the knowledge relevant to this context. The result has been a rather strong and sometimes vocal resistance by these users to learning "extraneous" knowledge about computers. IBM'S Job Control Language (JCL) is a good example of an awkward computer context interfering with the applications context which the analyst understands. The formulation of a working set of JCL commands is largely a heuristic process which involves a great deal of understanding about the computer context (more than applications people or managers care to know). Hence, the situation is quite common that once an analyst gets a JCL deck that works, that deck is passed around the office and everyone else copies it rather than try to learn the heuristics involved with formulating JCL commands.

The failure to define formally the procedural knowledge (especially the heuristic knowledge) needed to operate in the computer context has forced many departments to invest vast sums of resources in training their applications and management users in a secondary context. New techniques are needed which will allow those trained in the tools and support context to define the heuristics needed to access other types of knowledge which might be stored in the computers. The need to share computer context heuristics with others is increasing as it becomes more costly to train the users. The technology of knowledge-based systems (see Section 5.1.6) is already being applied to the problems of accessing a computer network and particular data bases.

	PROGRAMMING	SYSTEMS	OPERATIONS	...
FACTUAL	FORTRAN	Compiler theory	Computer site layout	
PROCEDURAL	Top-down design	Semaphore usage	Scheduling	
JUDGMENTAL	Use high-level languages only	Comparison of maxis vs. minis	Kill a job that is looping	

TOOLS AND SUPPORT/SKILLS CONTEXT

Figure 3-4

	MAINFRAMES	TERMINALS	COMMUNICATIONS	...
FACTUAL	Memory size; Execution speed; Nr. of registers	Nr. of function buttons; Character set	Line speeds; Line control	
PROCEDURAL	Operation via a stack architecture	Function button programming	Error checking and correction procedures	
JUDGMENTAL	Manufacturer's survivability	Merits of Incoterms vs. Computeks	Future of packet switching networks	

TOOLS AND SUPPORT/TARGET CONTEXT

Figure 3-5

3.2.3 DIRECTION CONTEXT

The context of direction (or management) spans both the applications and the tools and support contexts. Correspondingly, we identify three types of management: the management of applications, the management of tools (in particular, computers), and the management of other managers. As with the other knowledge contexts, the direction context involves both skills and target knowledge. Skills such as budgeting, personnel, and so forth generally apply to all forms of management regardless of the target being managed. The target, on the other hand, corresponds to an existing organization, project, or operation, and an effective manager will require specialized knowledge about that target in order to manage it properly. Figures 3-6 and 3-7 show some examples of skills and targets in the direction context. As with the mission context, knowledge about computerized tools can be useful to assist the managerial process, but the process of acquiring that knowledge can also detract from the manager's business of managing. The trade-offs must be evaluated. Section 6.2 examines the direction context through a Structural View of a department's knowledge resource and offers some suggestions about the kind of organizational structure which might be used to support a knowledge resource.

	BUDGETING	PERSONNEL	LOGISTICS	...
FACTUAL	Obligated funds	List of library books on personnel management	Industry standards (e.g., chair height)	
PROCEDURAL	Funds transfer procedures	Writing recommendations; MBO	Space template preparation	
JUDGMENTAL	O&M vs. RDT&E allocation	Promotion crieria	Sq. feet space requirement per employee	

DIRECTION/SKILLS CONTEXT

Figure 3-6

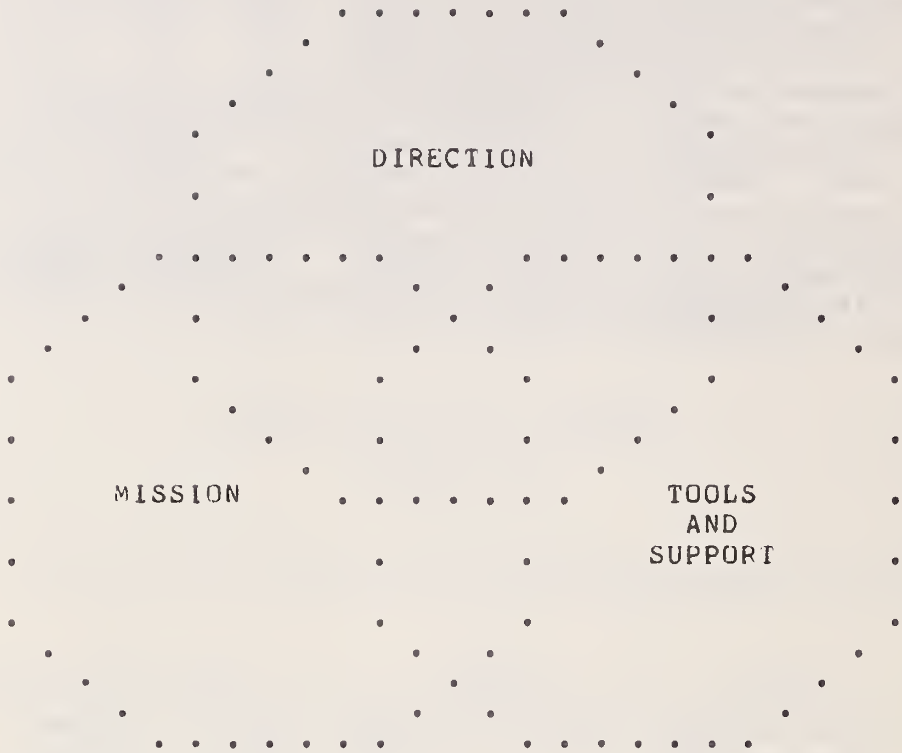
	DEPARTMENT	PROJECT	PROGRAM	...
FACTUAL	Amount of RDT&E dollars managed by department	Name of project manager	Program hardware space requirements	
PROCEDURAL	Procedure for reporting funds expenditures	Methods of acquiring people for the project	Development of space allocation	
JUDGMENTAL	Fund Project "B" vs. Project "C"	Identification of key people	Award of maintenance contract	

DIRECTION/TARGET CONTEXT

Figure 3-7

3.2.4 CONTEXT INTEGRATION

Figure 3-8 shows a Venn diagram representation of the three knowledge contexts relevant to this discussion. Where a particular piece of knowledge falls on this diagram is determined largely by the degree of relevance which that item has to one of the three contexts. The manner in which particular individuals access and use an organizational knowledge resource is the subject of the Functional View presented in Section 6.1. In that section, various job categories will be identified and placed on the Venn diagram in an effort to define clearly the role which each class is expected to play. The need for a coordinated understanding of the organization's knowledge resource should be obvious from this diagram. The acquisition of such a high-level view is the subject of Section 6.



CONTEXT INTEGRATION

Figure 3-8

4. ORGANIZATIONAL KNOWLEDGE RESOURCE

4.1 THE DEPARTMENTAL KNOWLEDGE RESOURCE

The authors contend that a department's knowledge, as described in Sections 2 and 3, is a valuable corporate resource which must be understood and handled effectively. In seeking to treat knowledge as a resource, an organization is actually seeking to formally organize its knowledge acquisition, utilization, maintenance, and preservation process. The emerging industry philosophy of treating data as a resource is really just a way of handling the acquisition and assembly of the factual knowledge which will be used by the various knowledge applications. Thus a data resource philosophy is based on the belief that it is sufficient to improve the processing of only the factual component of knowledge. A knowledge resource philosophy, on the other hand, is based on the belief that it is necessary to consider all of the knowledge of a Federal department and to include as much of it as possible in the knowledge resource. This is a recognition of the fact that a department's ability to supply the right knowledge (factual, procedural, or judgmental) to decision-makers (governmental and non-governmental) at critical times is a major determinant of whether the organization is judged to be a success.

Throughout history an organization's knowledge has resided in the heads of people: their understanding of the organization's goals, their understanding of what constitutes relevant factual knowledge, their facility with algorithms and techniques for problem-solving, their familiarity with the history of various problem areas, and their knowledge of skills and targets (including the ability of the people to keep up with changes in any of the areas). Traditionally, data has been passed from analyst to analyst on some form of storage medium (e.g., paper or magnetic tape), but other forms of knowledge have had to be taught. Associations among data items, together with the rules and conditions under which those relationships have meaning, have to be instilled in the minds of each analyst along with the reasons why this knowledge is important and why it should be preserved. The teaching of procedural knowledge is also important and it may become a long drawn-out process which needs many explanations and much on-the-job training. The teaching of judgmental knowledge is even more difficult. In any case, depending on the nature of the type of knowledge, this educational process can require many years and may consume a significant portion of an organization's resources. This situation is not unique to government, of course. In fact, certain studies indicate that approximately 20% of the U. S. Gross National Product is currently devoted to such knowledge-transferal activities [15].

It is becoming clear that the transmittal of knowledge from individual to individual within the government is growing increasingly expensive, for the amount of knowledge required for a department to function is growing at an ever-increasing rate. A complaint frequently heard from all levels of management is

that oftentimes it is next to impossible to contact the persons who have the relevant portions of the knowledge needed to answer a given question, to perform a specific task, or to solve a troublesome problem. Many times the knowledge of whom to ask is not even available. Furthermore, the cost of human storage (in the form of employees' brains) is also going up. Perhaps the most pressing knowledge-related problem concerns the departure from the government of many of our most respected and knowledgeable experts in fields such as transportation, agriculture, defense, health, commerce, etc. It is essential that departments somehow capture the unique knowledge which their key experts have before it is gone forever. Publications, reports, courses, textbooks, and protege-teaching all help, but a great deal of organizational knowledge is being "preserved" in the form of application programs as the computer is used to perform more and more data processing. Anyone who has attempted to read someone else's computer program code can attest to the fact that, currently, computer programs are an extremely awkward form for storing knowledge. The act of documenting programs is an attempt at writing down and preserving, in a form more readily understood by humans, the relevant knowledge from all three contexts of mission, direction, and support which the programmer had at the time the program was written. The current emphasis on program documentation (both line-by-line and global) in government computer centers and elsewhere is an open admission of the fact that knowledge represented in the form of computer program code is neither convenient nor sufficient for long-term use, given today's programming environment.

Improved programming languages and computational environments (e.g., ECL [19] might make the embedded knowledge more accessible, but there will remain the need for an overall plan regarding the organization's behavior with respect to the effective sharing of such knowledge. Such a policy must be based on a sound understanding of the organization's knowledge. A model which can be used to gain that understanding was presented in Sections 2 and 3. An organization must develop some formal mechanisms for knowledge representation, knowledge acquisition, knowledge application, and knowledge preservation. This is not to say that humans should be replaced by machines, but rather, that humans need better ways to cope with their knowledge problems. The difficulties lie, not so much in quantity of knowledge, (the amounts are doubling approximately every five to ten years) but in access and in understanding. Improved knowledge-access assistance is already being provided in the fields of medicine and chemistry where knowledge-based systems have begun to demonstrate remarkable capabilities. (More will be said on this subject in Section 5.) In any case, improved access to an organization's knowledge can allow organizational problem-solvers to profit from the collective experience of others in making their decisions by examining the rules and conditions which others have used to solve problems before them. Such capabilities could enhance the day-to-day operations of many Federal departments.

4.2 KNOWLEDGE MANAGEMENT

In Sections 2 and 3 we have described some ways of categorizing knowledge which are useful to understanding what is in an organization's knowledge resource. There are several aspects to the knowledge possessed by an enterprise and each of these different kinds of knowledge requires special consideration. The key to understanding this conglomerate of knowledge is the concept of knowledge independence. What is needed is a mechanism for maintaining each type of knowledge separately and independent from the other forms. In this fashion various pieces of knowledge can be used by different applications in a variety of situations with limited conflict. Application of the concepts of knowledge independence is an overt recognition of the existence of and the differences among the various types of knowledge which are each important to the mission of an organization. It is also an attempt to provide an environment conducive to knowledge-sharing throughout a department.

It should be emphasized at this point that a decision to treat knowledge as a resource has implications for the entire organization. It is likely that thousands of applications may need to access and use a knowledge resource. These applications are best understood and managed by the various technical specialties of a department (such as economists, physicians, scientists, statisticians, etc.) which use factual knowledge in their own specialized ways by applying different sets of knowledge rules and techniques (procedural knowledge). With the advent of a knowledge resource, specialists will continue to need to build their own sets of procedural knowledge. However, with a policy which pushes for a knowledge resource, the organization should begin to understand that there are many underlying generic problems and solutions which cross application boundaries. These general principles often have been overlooked with a concentration on a single knowledge application. (The processes of deductive inference are basically the same across applications and it is likely that one deductive inference tool could serve numerous applications). Indeed, past concentration on applications frequently has resulted in the binding of facts to the procedural knowledge of the application with a resulting unavailability of either the application's procedural knowledge or the application's facts to other users. Even when the data is available, the rules and conditions relevant to the data (i.e., the procedural knowledge) are not generally readily available. Many times the metadata and the procedural knowledge (if they exist at all) are buried in a morass of computer instructions implemented in Fortran or assembly language or some other obscure form. Attempts to use data without a proper understanding of what the data means can lead to misinterpretation and ill-advised decisions. Furthermore, data accumulated under these conditions has little or no long-term value for the organization since the cost of teaching lots of people all of the knowledge which is necessary to understand the data frequently exceeds the value which might be gained from that understanding.

The knowledge-as-a-resource philosophy attempts to separate all forms of knowledge in order to insure the availability of a given form of knowledge to the person or the application which might require it. A basic premise of this paper is that a knowledge resource should be provided to people or to various applications without dictating the manner in which the knowledge is to be used. Factual knowledge (in particular, data) is expected to comprise the largest portion of this resource which will be shared widely. The knowledge-as-a-resource philosophy implies that the concept of "ownership of shared data" must be transferred to the department as a whole in order to insure that the data is put to the broadest possible use and that other potential knowledge applications are considered before alterations which may affect the general utility of the data are permitted. One of the most critical instances of the conflicts which can arise occurs when a specific application wishes to delete data or relationships in which it no longer has an interest. If the data or the relationships are still being used by some other applications, dire consequences can result from the first group's autonomous actions. Some mechanism is needed to protect the interests of each organization in the department and to preserve the value of the knowledge resource.

In the final analysis, data has always belonged to the entire organization. At various points in time, it has been placed under the temporary custodianship of various people in the organization. From custodianship it was a relatively simple step to parochialism about the ownership of data. Individuals began to believe that data belonged to them and not to the organization. While some people have viewed this move as unexpected, it was almost inevitable once users began to embed procedural knowledge in the organization of their data. (For example, the order of records in a file may have some special significance for an application.) Since applications hold the purview over their procedural knowledge, they naturally extend that purview to the data. Knowledge independence (see Section 3.1) is the key to resolving the question of data ownership because the data can then be shared on a wide basis without the necessity for individual ownership. A department must begin to encourage users to concentrate on building and improving their procedural knowledge bases while taking vigorous steps to provide the data resource which these knowledge bases will need. Such steps might include

(1) adopting data base management systems which support the explicit use of metadata

(2) separating programs from the data which the employ

(3) adopting software engineering techniques to build shared software libraries of procedural knowledge

(4) adopting knowledge-based system technologies for limited reasoning capabilities coupled with access to large amounts of factual knowledge.

If data is to be shared among the various elements and knowledge applications of the department (where appropriate) then the data must belong to the department itself and not to some subdivision (or application). Organizational success in separating its factual and procedural knowledge is the key to this sharing. At the same time, organizational control must be exercised through some mechanism where decisions can be made for the good of the department as a whole, where priorities can be established and maintained deliberately, and where directories on the knowledge resource can be kept current. Such a mechanism can provide the foundation upon which the management of knowledge should be based.

Many new tools and techniques will be necessary in order to transform knowledge into a departmental resource. Many new and unusual problems may arise, and changes in managerial attitude and organization may be required. In the following sections we describe a methodology for implementing a knowledge resource in a department. We propose a system architecture based on the concept of maintaining knowledge independence which is geared toward providing a large degree of flexibility to both the end-users and the system designers. We also identify a set of functions which we feel will be necessary if a department is to develop knowledge into a departmental resource. Finally, in Appendix A-1 we offer a discussion of the kinds of research activities which will be needed to support a knowledge resource. One set of such activities which are already underway and which deserve special consideration is the topic of Knowledge-Based Systems. This subject is addressed in the next section.

5. KNOWLEDGE-BASED SYSTEMS

The technology of knowledge-based systems appears to hold great promise for providing many of the techniques and theories that will be needed to implement a knowledge resource. By a knowledge-based system we mean a system which employs sources of specialized procedural knowledge about a particular problem domain or problem-solving behavior in order to accomplish some task more effectively. Such systems differ from conventional computer programs (which have their procedural knowledge embedded deeply within the program) in that their knowledge is represented explicitly and separately from the code of the program which accesses and uses that knowledge. It is also important to note that knowledge-based systems deal with the processing of symbolic information as opposed to numerical computation. Such symbolic processing requires a different orientation of thinking about a particular problem area from the earlier "data processing" approach.

5.1 EXAMPLES

Knowledge-based systems have been developed to produce useful results in a number of diverse technical areas. Some of these areas include mathematical symbol manipulation (MACSYMA), human speech understanding (HEARSAY-II), chemical analysis of mass spectral data (DENDRAL), medical diagnosis of disease (MYCIN and DIALOG), signals analysis (HASP), computer network protocols and terrorist-tracking (RITA), and the representation and analysis of pictorial information (VIKING Mars probe). In the following sections we briefly describe these systems. Admittedly, neither the list nor the discussion is complete. Our intention is that these sections serve only to give an indication of the existence of such software and how it has been used. Further investigation of this technology by the various departments is suggested if they are to determine how the existing technology can be applied to their situations.

5.1.1 MACSYMA

Engineers and applied mathematicians generally could use computer assistance in non-numerical methods of analysis (e.g., symbolic integration). MACSYMA [20] is a knowledge-based system developed at Massachusetts Institute of Technology which performs a wide variety of symbol manipulation functions. Indeed, MACSYMA has already been used successfully by mathematicians over the ARPANET to solve difficult symbol manipulation problems.

5.1.2 HEARSAY-II

Speech understanding is an area of considerable interest to many segments of the government. The HEARSAY-II system [21] from Carnegie-Mellon University is a knowledge-based approach to this problem. The goal of the HEARSAY-II system is the understanding of continuous speech in sentence units. HEARSAY-II is of particular theoretical interest for its ability to employ knowledge

from a hierarchy of domains. The domains of knowledge which are relevant to speech understanding include waveforms, phonemes, syllables, words, phrases and sentences. In HEARSAY, knowledge about a particular subproblem in speech understanding can be used by modules attacking other subproblems to assist in their reasoning. For example, knowledge about the construction of words in spoken language may be used to assist the module concerned with the translation of waveforms into possible utterances. This type of approach might be used in knowledge-based approaches which need to correlate knowledge from a variety of sources and in a variety of forms.

5.1.3 DENDRAL

DENDRAL [22] is a system developed by researchers at Stanford University to aid chemists in the analysis of mass spectral data. DENDRAL has embedded within it sets of rules (see Section 5.1.9) which express many chemists' knowledge about mass spectrometry. Chemists are able to extend the system's knowledge by adding rules that apply to new classes of chemical compounds. To date, the results from DENDRAL analyses have been significant enough to be published in the chemical literature (rather than the computer literature). Even though only a few departments might be interested in mass spectrometry, the idea of identifying rules which can explain observed patterns in data is a classic form of analysis which can apply to many problem areas. In fact, the concepts embodied in DENDRAL have already begun to be applied to other problem domains.

5.1.4 MYCIN

The MYCIN system [23] was developed as a joint project by the computer science and medical departments of Stanford University and it is being successfully used to assist physicians with the diagnosis of disease and the selection of appropriate therapy for patients with bacterial infections of the blood. The knowledge base of the system contains about 200 rules on some fifty microorganisms and on approximately twenty antimicrobial agents. After being given some clinical data by a physician, MYCIN attempts to determine whether a patient has a significant disease. It then tries to infer the likely cause and, if possible, selects one or more recommended therapeutic agents. There may be a direct analogy between this type of medical diagnosis and many other types of diagnosis which government analysts perform daily. The researchers at Stanford have recognized this fact and they are working on a MYCIN-like system which can be used in non-medical problem domains.

5.1.5 HASP

Signals analysis functions performed in different departments may find a direct analog in the HASP system [24] at Systems Control, Inc. HASP has been designed to perform analysis on signals obtained from multiple sources and has been used to predict patterns in incoming data. Like HEARSAY-II, HASP is capable of

correlating knowledge from a variety of problem areas. The developers of HASP believe that it will be possible to produce a HASP-like system which will work in other problem domains.

5.1.6 RITA

RITA [25] is a system which is based on the MYCIN work but which was developed by the Rand Corporation to run on a PDP 11 minicomputer. A computer network protocol application has been developed which provides a generalized interface to the ARPAnet. This work is thought to be directly transferable to any ARPAnet-like network which has an associated terminal subsystem which can run RITA. Some specialized knowledge about unique eccentricities of these different networks may be needed, but, in general, the techniques should be similar.

A terrorist-tracking application on RITA would also appear to be relevant to government agencies which have been charged to protect the United States from terrorists' activities. In this application the system has knowledge on various terrorists groups, and, given an event, RITA attempts to infer which group was responsible for the act, what the group's goals are, and which negotiator would be the best person to deal with this group.

A third application of RITA is as an interface to the New York Times Data Base. The New York Times Data Base uses a complex and confusing (to a naive user) protocol and syntax for its queries. RITA has rules which specify how correct syntax for queries is produced and the system is able to use its rules to generate queries which the Times Data Base system will accept. Similar activities of interfacing to other data base management systems would also appear appropriate. In fact, RITA has already been linked to one data base management system (INGRES) in an early experiment.

RITA exhibits two forms of inferencing capability: goal-oriented and event-driven behavior. The goal-oriented mode is similar to MYCIN's deductive capability where the system is given a goal state and requested to determine how that state can be achieved (e.g., the terrorist-tracking application). The event-driven behavior is exhibited in the ARPAnet interface application where the system is given a task to do and then proceeds to accomplish that task by interacting with the environment.

5.1.7 DIALOG

The DIALOG system [26] developed by the University of Pittsburgh is a diagnostic consultation system for internal medicine. It has demonstrated a capacity for solving difficult clinical problems which are complicated by the concurrence of several disease entities. A major reason for its success is due to the fact that its knowledge base contains rules on something in excess of 50% of all known major diseases of internal medicine. The rule sets about the behavior of known types of internal

diseases have been formalized and placed into disease hierarchies. DIALOG uses these rules to form hypotheses. For example, if DIALOG has determined (based on primary clinical data) that a patient has congestive heart failure then the next problem which it attempts to solve is to differentiate between pulmonary emphysema and bronchial asthma (both cause congestive heart failure). Supportive evidence is required to confirm one of these hypotheses. Once this differentiation is done, the system will proceed to work on increasingly more specific hypotheses which might be related, for example, to pulmonary emphysema. DIALOG is of particular interest to government users because of the style of inferencing which it employs. This type of inferencing is called abduction (as opposed to the more familiar forms of deduction and induction). Abduction is the form of inference which starts with a set of rules about certain sets of facts and guesses that a new given fact is a case under a particular rule in the set. DIALOG also has the ability to function with a sizeable knowledge base (10,000 rules). Undoubtedly, as an organization becomes more sophisticated in their use of knowledge-based systems the size and complexity of their knowledge bases will grow. It is not unreasonable to envision government applications which require knowledge bases of this magnitude.

5.1.8 VIKING MARS PROBE

Cal Tech's Jet Propulsion Laboratory (JPL) in association with NASA has employed a knowledge-based system in the perception subsystem of the VIKING Mars probe vehicle prototype [27]. This knowledge-based system is designed to perform an analysis of pictorial information of the terrain in the landing area of the VIKING Mars probe. Since NASA had never been to Mars, it was difficult for them to predict exactly what Mars' terrain would look like. What will a picture of a rock on Mars appear to resemble, for example? In any case, there is quite possibly an application of this technology to other government interests in the processing of pictorial information.

5.1.9 PRODUCTION SYSTEMS

Most of the knowledge-based systems mentioned above fall into the category of Production Systems. Knowledge is represented in these systems in the form of productions or rules which state that if a certain event is true in the system's data base, then a certain action should be taken. This action may, for example, alter the data base, or it may perform some computation. Such systems may be environment-driven in which actions occur as a result of conditions arising (as in the RITA network protocol system), or they may be goal-driven in which backward-chaining through the rule set is employed to determine the set of conditions needed in order for a certain action to take place (as in MYCIN's diagnosis process).

5.2 KNOWLEDGE ISSUES

There are several important areas of consideration in knowledge-based system application. These areas include knowledge representation, acquisition, and utilization, inexact inference, and explanation. Various knowledge-based systems offer different techniques for representing, acquiring, and using knowledge as well as methodologies for handling inferences under uncertainty and explaining to the user why a particular conclusion was reached.

5.2.1 KNOWLEDGE REPRESENTATION

Knowledge representation deals with ways of representing knowledge to the system. These representations include factual knowledge (such as those based on text-book data), well-tooled models (such as laws of physics), and heuristic knowledge (or rules-of-thumb based on experience but without formal proof).

5.2.2 KNOWLEDGE ACQUISITION

Knowledge acquisition concerns the process of extracting knowledge from experts and transferring it to a form which the system can use. Most of today's knowledge-based systems require this knowledge acquisition to be done by hand over an extended period of time, although some systems (e.g., METADENDRAL [28]) do attempt to extract knowledge from sample data, i.e., they exhibit some signs of primitive learning.

5.2.3 KNOWLEDGE UTILIZATION

The issue of knowledge utilization is, perhaps, of more interest to the end-users, for it is at this stage that the knowledge is put to use. Some systems (notably HEARSAY-II) attempt to bring together knowledge from different sources to solve a given problem. The value in bringing together different kinds of knowledge is one of the reasons why complex government agencies such as ERDA are created.

5.2.4 INEXACT INFERENCE

Inexact inference refers to the problem of making inferences from incomplete or probabilistic knowledge. This situation is probably extremely prevalent in most government departments since the nature of their business is to piece together knowledge from a collection of facts gleaned over time. Systems such as MYCIN can associate degrees of certainty with individual rules and thereby produce responses of the form "The answer is most likely (.7) ..."

5.2.5 EXPLANATION

A very useful aspect of many knowledge-based systems is the ability to explain the system's line of reasoning to a doubting or novice user. Such a capability is essential when critical

decisions are to be made or when (as will often be the case) the knowledge of the system is to be acquired and to grow incrementally. Explanations can be an integral part of the users' learning with systems of this type.

The technology of knowledge-based systems is an active research area which appears to have good potential for implementing specific knowledge applications in government departments. Second generation knowledge applications are now being developed as the technology matures. What is needed now is a study of the state-of-the-art in knowledge-based systems and an assessment of the suitability of potential government applications for this technology. The larger problem of knowledge-sharing among various applications will require extensive planning, thinking, and experimentation as each department moves toward developing their knowledge resource. The next sections of this paper deal primarily with ways to view a knowledge resource which can lead to the development of the proper perspective for the implementation of a knowledge resource. These views form the basis for the logical system design of Section 8.

6. KNOWLEDGE RESOURCE VIEWS

There seem to be at least three ways to view the knowledge resource in large, complex enterprises such as the Federal Departments: a Functional, a Structural, and a Physical View. The Functional View emphasizes the various kinds of user-functions performed within the enterprise regardless of the equipment or organization involved. The Structural View describes the organization which the enterprise uses to establish, build and use its knowledge resource. The Physical View depicts the physical or computer environment in which the knowledge is processed. These three views are not intended to be mutually exclusive; rather, they are intended to provide three different models of the knowledge resource of an enterprise. The first view identifies various needs of each functional activity, thereby permitting optimization by specialization of function. The second view emphasizes the organizational roles which people play in operating and using the resource. The third view introduces the physical constraints imposed by the technology which exists in the real world and, at the same time, points out areas of weakness requiring additional acquisitions or development on the part of the enterprise. These three views are useful in studying how a department can best prepare for the business of handling knowledge so that it may become an organizational resource.

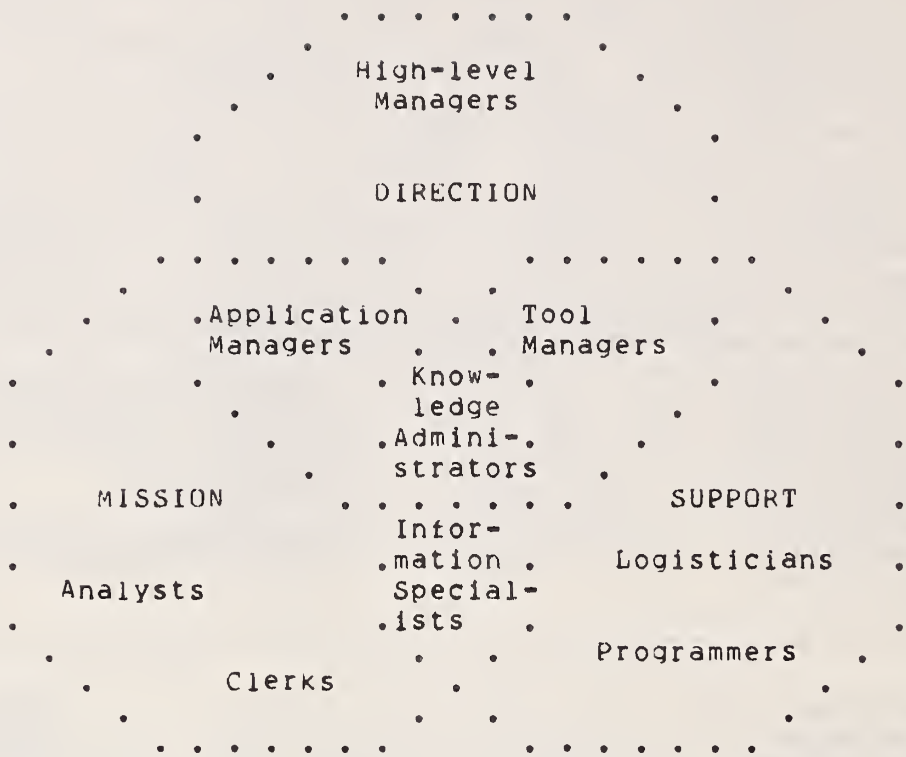
6.1 FUNCTIONAL VIEW

From the point of view of functions, the knowledge resource of an organization must service several different kinds of users (i.e., it must serve many different functions). We find it convenient to identify six general classes of users of a knowledge resource: Clerks, Managers, Analysts, Information Specialists, Programmers, and Knowledge Administrators. Undoubtedly, other classifications are possible. Figure 6-1 shows a grouping of these functional users according to their degree of sophistication with a computer environment as opposed to their sophistication with the applications which may require some programs to be run on computers. In Section 3.2 we discussed three knowledge contexts which are of particular relevance to a department: the mission context, the tools and support context, and the direction context. The various functional categories have a very definite role with respect to these contexts. The Venn diagram of Figure 3-8 is reproduced in Figure 6-2, and the various functional categories are superimposed over it. Of particular interest are the points of interaction among the three areas of context, for it is at these points where knowledge-sharing between contexts will cause the most difficulties.

```
 Computer                                            Computer
 Naive                                               Sophisticated

<------------------------------------------------------------------>

   +----------+                         +-------------------------+
   |Clerks,   |                         |Information Specialists,|
   |          |                         |                         |
   |Managers, |                         |Programmers,             |
   |          |                         |                         |
   |Analysts  |                         |Knowledge Administrators|
   +----------+                         +-------------------------+

<------------------------------------------------------------------>

 Knowledge                                           Knowledge
 Application                                         Application
 Sophisticated                                       Naive
```

FUNCTIONAL ORIENTATION

Figure 6-1



PEOPLE ROLES VS. CONTEXT

(Relationship of Functions)

Figure 6-2

"Clerks" are heavily applications-oriented with specific and somewhat repetitive tasks. Their application-orientation is reflected in Figure 6-2 by their placement entirely within the mission context. Personnel in this category might include census takers who are employed by the Bureau of the Census to gather the census data. The personnel at the Veteran's Administration who prepare the forms which are used to certify that a veteran is eligible for benefits is another example of people who fall into this category. Primarily their function is to collect data for the department and to make note of special activities for storage in the knowledge resource. They employ algorithmic knowledge in an applications context as an integral part of their job to provide information and to note transactions, but they have very little understanding of the complexities of the other knowledge activities which occur as a result of their actions. They usually have their expertise in a limited subject area. For example, a census taker might be very skilled in getting the census forms filled in, but he or she might not have any idea of how to analyze the data which they have collected. Users in this category require conceptually simple operations which reflect the algorithmic nature of their jobs. They also require rapid response to their commands because of the transaction nature of their interactions.

Under the category of "Managers" we have grouped all levels of decision-makers and planners for the department. Three types of managers are identified: those who manage applications, those who manage computers or other tools, and those who manage other managers. The first two categories require a blend of knowledge from the management context with either the mission or the computer-tool context. This is shown in Figure 6-2 by placing these two types of managers into both contexts. The Managers group generally uses judgmental knowledge in large quantities and views the knowledge resource from the direction context. Their function is to receive and analyze requirements, to set goals for the department, and to see that things run smoothly in attempting to meet those requirements. Members of the Managers group employ knowledge in a largely heuristic fashion to assist in making decisions and to experiment with hypotheses (i.e., to play "what if" games). From the subject area perspective, applications managers have knowledge ranging over a broader area than either clerks or analysts. Similarly, computer managers generally have a broader knowledge of the computer context than do programmers. A goodly portion of their knowledge is used to achieve proper resource assignment in this context. Users in the management category need "natural" high-level operations which provide "knowledge-rich" responses to a brief request. Due to (1) the limited amount of time which the members of this group generally have, (2) their relatively high cost, and (3) the heuristic nature of what they do, training must be be kept at a minimum, and commands and protocol must be easily recalled.

"Analysts" refers to the large category of government employees whose knowledge context consists of a basic mission skill combined with knowledge about a specific geopolitical target or

subject area. Figure 6-2 shows analysts contained within the context of their application. They are highly trained in a knowledge specialty other than computer or information science. Typically, these are the specialists who develop the algorithms or the heuristics which are used to process factual knowledge. Part of their knowledge is contained in the algorithms and heuristics they employ. This is, of course, the ideal case. In actual fact, many analysts may have been cross-trained in many of the skills of the computer context. Some example members of this group might be economists, statisticians, transportation specialists, nuclear physicists, etc. Analysts are the most expert people in their knowledge application area, but their expertise does not usually extend outside of a given knowledge skill area. For example, knowledge of nuclear physics may not help a person decide on the advisability of building a coal gasification plant. However, analysts will need and use target knowledge (energy needs of the United States in 1985-1995) which is common to many other applications skills. Since this group uses algorithmic and heuristic knowledge against the data bases and the metadata, they have broad interface requirements. Characteristic of the members of this class is the need to access heuristic tools and a need for flexible interfaces for accessing the knowledge resource. They also require a set of powerful but explicit operations which allows them to state algorithmic requests and interpret the responses precisely.

"Information Specialists" are, for the most part, retrieval specialists who provide an information service to the clerks, analysts, and managers. People in this category are generally highly trained in the various retrieval mechanisms (i.e., "tools") which can be used to access the data bases and the metadata. Therefore, they need to have some knowledge of the computer context. They also need to receive some training in the particular specialty areas of the people whom they are supporting. This training enables them to provide service in the use of algorithmic knowledge which has been implemented in computer program form. Information Specialists, in effect, act as human translators of people's requests into machine-oriented queries and, to some extent, assist in the interpretation of the results. Information Specialists require access to all the knowledge resource facilities available (except, perhaps, the heuristic facilities). They will need extensive and on-going training as new information processing tools are introduced and as new data bases are implemented.

The fifth category is "Programmers", whose specialty is computer programming. They, of course, view the knowledge resource from the computer portion of the tools and support context as shown in Figure 6-2. Programmers provide an oftentimes necessary service to other functional sets of users where precise, repetitive algorithms are required. Users in this category need to become more intimate with the internal workings of the underlying software which manages the data bases and the metadata. They are also frequently called upon to code the algorithmic knowledge of certain application or management skills into computer programs.

This latter need means that they require very formal, very precise interfaces to programming languages in order to carry out their function.

The final group is termed "Knowledge Administrators". This group consists of three kinds of people, the Enterprise Administrator, the Data Base Administrator(s), and the Applications Administrators, whose collective function is to maintain an efficiently running and effective knowledge resource. A detailed examination of the Knowledge Administrators occurs in Section 6.2. The central role of the Knowledge Administrators is depicted in Figure 6-2 which shows the members of this category as dealing with all three contexts. Knowledge Administrators are less interested in obtaining specific facts from the knowledge resource and more interested in monitoring how the knowledge resource is performing and meeting the needs of the other user groups. In this capacity they will require a special set of monitoring and statistical evaluation tools.

In order to support the continuum of user functions, three broad categories of software must be considered: (1) the data management software which provides the "access engines" to data bases contained on some storage medium, (2) the procedural-oriented software (i.e., algorithmic and heuristic processes, and (3) the user interfaces which must be capable of providing the widely disparate knowledge application views of the users in the six functional groups. This categorization highlights two diametrically opposed aspects of information systems: machine efficiency and user efficiency. The three categories should be as independent as possible, although most current commercial software tends to tie the user interfaces closely to individual data managers or algorithmic procedures. Greater independence of the categories allows the users to view the data in ways which are natural to them and are closely tied to the particular knowledge application without forcing them to understand how or why choices of machine efficiency have been made.

6.1.1 KNOWLEDGE RESOURCE TOOLS

The software tools which can be used to build and access a knowledge resource can be grouped into several categories which differ from each other primarily in the degree of logical structuring and independence of knowledge which each system can accommodate. Figure 6-3 shows a continuum of knowledge resource tools running from the standard file system supplied with a computer's operating system, through file management systems and data base management systems, to knowledge-based systems, and finally to a new kind of system (to be defined below) termed a knowledge base management system. These five kinds of systems differ in their purpose, their function, their capabilities, and the view of data and metadata which they provide to users. They also differ in their ability to support the needs of a knowledge resource (the independence of knowledge contexts and knowledge forms). To support systems which are intended to be knowledge-based, the data base must contain facts and assertions about the world, must

provide for relationships (metadata), must be accessible by a wide range of knowledge applications, must be of arbitrary size, and must have no a priori constraints on the complexity of organization.

Many varieties of the knowledge resource tools described in Figure 6-3 are currently available, and, in fact, there are several such products in different departments today. The use of these tools within a department requires a considerable learning effort for each tool. Unfortunately, there is no guarantee that a department will be able to get these tools to function together. What we are proposing is a framework for tying those various systems together so that the knowledge which they manage can be shared most effectively among themselves and with the human users which they serve. The philosophy of treating knowledge as an organizational resource recognizes the need for such coordination, and any methodology for implementing a knowledge resource must take this aspect into consideration.

```
    Physical                                              Logical

<--------------------------------------------------------------->

+--------+  +-------+  +----------+  +----------+  +----------+
|Standard|  |File   |  |Data Base |  |Knowledge-|  |Knowledge |
|File    |--|Manage-|--|Management|--|Based     |--|Base      |
|System  |  |ment   |  |System    |  |System    |  |Management|
|        |  |System |  |          |  |          |  |System    |
+--------+  +-------+  +----------+  +----------+  +----------+

No          File/      Inter-record  Rules of      Knowledge
Structure   Record/    Relationship  Inference;    Independence
            Field/     Description   Conditions
            Descrip-                 for Applying
            tion                     the Rules
```

KNOWLEDGE RESOURCE TOOLS CONTINUUM

Figure 6-3

The standard file system (SFS) provides the basic access methods employed by the other, higher-level data managers. The SFS can provide files directly to users (generally to programs or to a text editor), but they require the user programs to contain much of the information about the physical characteristics of the file and all of the information about the logical structure of the data. The file is, in effect, a sequence of bits which is meaningful only to the knowledge application which created it.

A file management system (FMS) begins to allow rudimentary data description in the form of files, records within files, and fields within records. Many commercial packages are available which fall into this category, including some stand-alone inverted-file retrieval systems. An FMS is designed to process groups of related data in a fast, cheap fashion where separate logical relationships among data aggregates (records) are not necessary. Such relationships are again retained by the programs which access the data (which means that the knowledge is not readily available to other algorithms or heuristic procedures).

A data base management system (DBMS) maintains and provides access to an integrated data base of many interrelated data aggregates. Provision is made for logical structuring of the data to reflect both abstract and physical relationships. This structure is made explicit in the data base itself (in the form of metadata) and is available to the DBMS. This permits access to the description of an entity (such as by listing its attributes or characteristics) independent of the actual occurrences of entries for that object in the data base. For example, commodity sectors of agriculture can be divided into crop and livestock classifications. Crops can be further divided into fibers, tobacco, grains, oil crops, fruits, vegetables, ornamental plants, and sweeteners. The attributes of the commodities can be listed independent of any particular commodity which is represented as an instance in the data base. Furthermore, an integrated data base permits the expression of multiple abstract relationships among data aggregates so that a department can describe not only what an entity is but how it is used by the different agricultural subsectors within the economy. Continuing the example, relationships between supply and demand with respect to each of the commodities need to be recorded. Ratios between production costs and distribution costs are also important and need to be noted. This type of factual knowledge could be used by pork distributors, for example, to plan an advertising campaign which might be designed to offset an abundant supply of pork. A data base management system permits an integrated representation of data in support of all of these potential uses.

A knowledge-based system (KBS) is a term used here to represent a new kind of experimental processing system which is not yet available in the commercial marketplace (see Section 5 on Knowledge-Based Systems). A KBS allows for the representation of rules of logic and conditions for the application of these rules to be placed within the knowledge base and accessible to the KBS. The KBS needs a data base and metadata against which to apply its

rules. Most current implementations include a specialized DBMS built for the particular knowledge-based system. In this regard, the factual and procedural knowledge are not clearly separated, thus making it difficult to share them with other applications.

A knowledge base management system (KBMS) is a term which we have coined to represent the extension of the knowledge-based philosophy to include the concept of knowledge independence. The concept is a rather straightforward extension of the data base philosophy which attempts to extract the logical structure of the data base and keep it separate from programs (procedurized applications knowledge) which access it. Formally organizing and representing knowledge in this fashion (separate from the applications which employ it) is a positive step toward preserving this knowledge in a formal manner for future generations of users. A knowledge base management system employs all of the other types of knowledge software tools in a consistent fashion striving to maintain the independence of the various forms of knowledge. A KBMS is the foundation upon which a department can build and manage its knowledge resource.

Each of these kinds of systems provides a special service to the enterprise and each has certain advantages and disadvantages. SFS and FMS violate the knowledge independence goals upon which a knowledge resource is founded. Furthermore, few people in the mission or direction contexts prefer to think of their factual knowledge in terms of strings, fields, records, or files. However, the claim has been made by those in the computer context that these systems provide higher degrees of processing efficiency. In general, as one moves from the SFS to the KBMS, one presumably sacrifices some ability to tinker with processing efficiency while gaining some degree of programming or human efficiency through knowledge independence. The trade-offs are not always clear, but in general they hold true. Not all applications, however, can afford a full-fledged data base management system. Designers of single-user systems with non-shared files or special-purpose real-time processing systems may find that the overhead involved in the generality of a DBMS is not cost-effective for their application. Presumably, then, an FMS or the SFS would suffice by providing the necessary speed at the cost of increased programming and maintenance effort (since the procedural and factual knowledge will quickly become hopelessly intertwined). Once this happens, of course, the data will be closed for all other knowledge applications. This point is a legitimate part of the life cycle cost of a software system, but one which is rarely measured in today's environment.

Many times the question is asked (especially in large system development projects), "What is the best DBMS for my system?" The answer often comes back after considerable study and evaluation "None; but system XYZ appears to be the least bad." What has happened, of course, is that no one DBMS (or FMS or SFS) meets all of the requirements of the various knowledge applications of the proposed system. Some knowledge contexts may need fast retrieval but medium update speeds. Others may need

complicated data structures. Still others may need to store "live" data very fast and then process it later in background mode. The key point is that the system designer should have available a choice of which kind of software is most appropriate for the individual applications which the system is to serve. No single knowledge resource tool can properly service all users. Beyond the variety of software, then, some mechanism will be needed for overseeing the usage of these various knowledge tools and coordinating any interaction which may be necessary among them (more about this control mechanism will be presented in Section 8).

In attempting to move to a position where knowledge becomes an organizational resource, a department will need more than a selection of data management software; they will need a formal mechanism for maintaining their heuristic knowledge apart from their applications software (which represent algorithmic knowledge). As discussed earlier in this paper, knowledge is more than structured data. It includes the semantics of the data, how the data is to be used, and the representation of why the data is important to the department. In a knowledge resource which is meant to be relatively complete, some means will have to be provided so that non-computerized knowledge may be represented or accounted for in the system. For example, the answer to a query of a KBMS might be a list of data bases or files where the appropriate data may be found, but it might also be a location in a filing cabinet in the office, or it may be the name and phone number of the resident (human) expert who is quite likely to be the best available knowledge source for certain problems. With most present-day data managers, this association function is performed by humans who quite frequently are unavailable at the time the question arises. Granted, some systems may maintain a file of key personnel, but the knowledge of why those people are key or what questions they are likely to be able to answer is generally retained in the users' heads (where it can be hard to access).

6.1.2 USER INTERFACES

Just as there is the need in large-scale systems for several knowledge resource tools, each serving a different knowledge function, so, too, is there a need for several user interfaces which provide specific knowledge-based capabilities and which permit knowledge applications views of the data to the end-users. We believe that it is both desirable and necessary to distinguish among the human users of any information system in terms of their needs, desires, capabilities, motivations, and especially in terms of the needs dictated by the particular application knowledge of the user. In this regard, there is the need to develop user interfaces not only with an eye on what it might take to train the potential users, but also with an appreciation for preserving the view of data which is natural to them and to their application. This is really nothing more than a recognition of the fact that in today's world, and in the world of the foreseeable future, the main costs in an information system will

be for people and the creation and maintenance of their knowledge bases, not for machines to process them. With this fact in mind, we infer that there must be a rich variety of user interfaces available to support the wide variety of humans who are expected to use the knowledge resource. Figure 6-4 shows eleven different user interfaces which we feel are important for the proper functioning of a department. This list is not necessarily complete, depending on the user population and purpose (indeed, James Martin [29] has identified some eighteen distinct user interfaces). Further details about user interfaces can be obtained by referring to Martin's work, and the reader is encouraged to do so.

No one user interface is likely to be adequate for all users or all knowledge areas within the department. Each provides a specific view to accomplish a specific function. Neither must any one system provide all possible user interfaces. Only those interfaces which fulfill a real need should be included in a given system being developed for a project. For example, in a project where users employ algorithmic knowledge predominately, the users might be very content with a forms or a menu type of interface. On the other hand, a user who employs heuristic knowledge heavily would undoubtedly chafe under such restrictive interfaces. What is recommended is that a selection of user interfaces (standardized throughout the department) be available for system designers to pick and choose from as needed.

USER INTERFACE	DESCRIPTION
* Transaction	- Rapid response; terse commands; intended for all-day repetitive use; limited view of the knowledge resource.
* Menu	- Commands selected by the users from a list presented dynamically; very simple; primarily for casual users.
* Forms	- Data entry and retrieval is accomplished via predescribed forms; for casual and dedicated users both.
* Graphical	- (a) Information as charts, graphs, etc.; (b) Information as dynamic animation; easy to interpret; information-rich.
* Dialogues	- User-specific, interactive question/answer mode initiated by either the system or the user.
* Natural Language Subset	- A limited-context capability for specifying commands in a subset of natural language (e.g., English).
* Query Language	- A formal language for specifying requests explicitly.
* Relational	- A more powerful form of a query language which allows dynamic restructuring of data bases as needed.
* Programmatic	- Extension of existing formal computer programming languages to accommodate data management functions.
* Navigational	- (a) Access-path dependent traversal of a data base; (b) Access-path independent traversal of the knowledge resource.
* Text Editor	- To create and maintain local files of private interest (such as notes, debugs, and memos) as well as textual documents.

USER INTERFACES
Figure 6-4

Figure 6-5 indicates the authors' feelings about the relationship of the six user functional groups to the eleven user interfaces discussed in this section. There will be considerable overlap of individuals within the groups; consequently, it is not possible to limit the user interfaces to only a subset of the groups (or vice versa). However, in general, the primary expected usage is that shown in the figure.

It is evident from Figure 6-5 that Analysts and Information Specialists have need for very similar user interfaces. This is to be expected since their functions are similar. An Information Specialist is really just an Analyst with a concentrated specialty in the information tools available with the knowledge resource. It also appears from Figure 6-5 that some job skills (Analysts and Information Specialists, in particular) require a large number of user interfaces. It must be pointed out that while the functional groups may need all of the interfaces listed, no single person is expected to need to use more than a few. Even so, it might be useful to develop a single higher-level user interface from which each of the interfaces in Figure 6-4 can be accessed in a smooth consistent fashion. This high-level interface may or may not be identical for all functional user types. Such an interface is another potential application of a knowledge-based system (see, for example, Section 5.1.6 on RITA's high-level interface to the ARPAnet).

	Clerks	Managers	Analysts	Information Specialists	Programmers	Knowledge Administrators
Transaction	x					
Menu	x		x	x		
Forms	x	x	x	x		
Graphical		x	x	x		x
Dialogues		x	x	x		x
Natural Lang.		x	x	x		x
Query Lang.			x	x		x
Relational			x	x		x
Programs					x	x
Navigational			x	x	x	x
Text Editor	x	x	x	x	x	x

USER INTERFACE/FUNCTIONAL VIEW

Figure 6-5

6.2 STRUCTURAL VIEW

The Structural View provides a way of looking at the organizational functions which will need to be performed by a department which is seeking to manage its knowledge as a resource. This Structural View is based on the architectural model that was recently presented in an American National Standards Institute (ANSI)/X3/SPARC study group report on data management systems [30]. Their architectural model is designed to support the current and future requirements of large complex organizations and appears to provide parts of what would be needed to support a knowledge resource.

A short description which deals with the macro-level details of the architecture is followed by a chart which serves to capture the flavor of the architecture. Additional details may be obtained by referring to the source document referenced in the paragraph above, and the reader is encouraged to do so. In essence, the ANSI architecture is based on the notion that there are three realms of information that are of interest to an organization, namely: the real world, ideas about the real world that exist in the minds of its people, and the symbols on some storage medium that represent these ideas. The ANSI group believes that information in each of these realms has properties that differ subtly and significantly from those of other realms. These properties and their differences are used to form the basis for their information model.

The ANSI group postulated three models of the information world of an enterprise and designed an architecture to support the three realms of interest. The three realms are identified as:

External - a simplified model of the real world as seen by an application or family of applications

Conceptual - a limited model of the real world as viewed by the entire enterprise

Internal - a model depicting the data that resides in the physical storage medium.

Each of these realms consists of a data model and a schema describing that model. The objects of interest in the real world are called entities, and the facts of interest about the entities are called attributes. These concepts are consistent with our use of the terms as expressed earlier in this paper.

The external model is a collection of objects that represent the entities and attributes of interest to a specific application or family of applications. Each external model is described by an external schema. It follows that there will be as many external schemas as there are applications which use the data resource. The humans who are responsible for creating and

maintaining the external schemas and for providing the interfaces to the end-users of the data resource are called Application Administrators (AA).

The conceptual model is a collection of the objects that represent the total entities and attributes of the enterprise. (We shall expand this definition in Section 8 to include the knowledge of an enterprise.) The conceptual schema is created and maintained by a human called the Enterprise Administrator (EA). The main focus of this schema is on the organization's total knowledge resource. For obvious reasons there can be only a single conceptual schema in the department. It serves as the information model of the total organization.

The internal model is a collection of objects containing the stored data that represent the external and conceptual models. The internal model is described by an internal schema, and for consistency, there must be only one internal schema for each physical data base. Each internal schema is created and managed by a human called a Data Base Administrator (DBA) who is responsible for providing the interface to the hardware and software of the information system. Thus, the DBA must have a heavily technical orientation.

A macro view of the overall model is presented in Figure 6-6. As noted previously, the ANSI group identified different orientations for the conceptual, the external, and the internal parts of the overall model. They have correspondingly identified three distinct human roles which reflect the different orientations. The Enterprise Administrator deals with a high-level macro view of the enterprise's knowledge resource. The Applications Administrator deals with end users and their localized view of the knowledge resource. The Data Base Administrator deals with the highly technical hardware and software functions of the data base management systems and thus holds a physical view of the knowledge resource.

In one sense, the conceptual model of the Enterprise Administrator can be thought of as a top management or corporate view of the entire knowledge resource from the standpoint of the knowledge that is being produced and maintained and the data which the enterprise needs to acquire in order to support its knowledge applications. The EA seeks to optimize the acquisition and preservation of knowledge across the entire organization and must therefore be prepared to deal with all of the knowledge contexts of the department. The external model of the Applications Administrator focuses attention on the development and use of the application knowledge bases. The AA must be aware of the data bases and the metadata bases which the knowledge applications need to access in order to operate. This fact causes the AA to use a mission or application context when viewing the knowledge resource. The internal model of the Data Base Administrator centers on the data bases and metadata bases and on the technical details of providing that data to users rapidly and cheaply, and this results in a computer context orientation for the DBA. In

actual practice, the division of interests will not be nearly so clear cut, and the department will require cooperation from all three administrative functions as well as all three knowledge context areas.

The Structural View presented here will be used in the sections which follow to suggest the basic functions which we believe will need to be performed if an organization is to create, build, and maintain a knowledge resource. Support for the functions identified here will be contained in the logical system design of Section 8. The Structural View will also be used in Appendix-1 to identify problem areas and research topics which are relevant to the development of a knowledge resource.

```
                     +-------------+
                     | Enterprise  |
                     |Administrator|
                     +-------------+
                            |
                            |
                      +----------+
                      |Conceptual|
                      |  Schema  |
                      +----------+
                            |
                            |
       ---------------------------------------------
       |                                           |
       |                                           |
+-------------+                           +-------------+
|  Data Base  |                           |Applications |
|Administrator|                           |Administrator|
+-------------+                           +-------------+
       |                                           |
       |                                           |
  +--------+                                  +--------+
  |Internal|                                  |External|
  | Schema |                                  | Schema |
  +--------+                                  +--------+
       |                                           |
       |                                           |
+-------------+                                +-----+
|Data Managers|                                |Users|--+
+-------------+                                +-----+  |--+
       |                                          +-----+  |--+
       |                                             +-----+  |
 +----------+                                           +-----+
 |Data Bases|
 +----------+
```

ANSI/X3/SPARC DATA MANAGEMENT MODEL

THE STRUCTURAL VIEW

Figure 6-6

6.2.1 KNOWLEDGE RESOURCE FUNCTIONS

Some basic functions will need to be carried out if a department is to implement and maintain a knowledge resource. We have designed a theoretical structure which contains all of the appropriate functions in a single centralized organization. This is not meant to imply that centralization of the functions is the only possible solution or even the best solution. A centralized approach is presented here as a convenient way to identify and discuss the functions at one time. The actual implementation of an organizational structure for managing the knowledge resource is best left to the discretion of the individual departments. Figure 6-7 shows a possible break-down of tasks by function within the theoretical office of Enterprise Administration. The key executive officer in such an organizational structure would be an Enterprise Administrator. An important part of an EA's responsibility would be the direction and control of the knowledge resource center which acts as a nerve center to coordinate the knowledge activities of the department (see Appendix-1). Policy, Security, and Standards are important functions which are also represented in the diagram. Control and coordination of these efforts are needed to insure that proper enterprise goals are established and met. The function of the External Information Sources group is to integrate into the knowledge resource factual knowledge which is primarily textual such as documents, periodicals, newspapers, books, reports, and other library materials. These sources may or may not be automated. Knowledge Resource Technology Assessment incorporates the functions of tracking research efforts in the areas of hardware and software relevant to knowledge resource tools and assessing the utility and impact of such new technology on the organization. The technical role of the Data Base Administrator has been described earlier in the paper. The DBA's are located under the control of the EA in order to insure cooperation among the various DBA's toward the common goals of the department. Note that there may need to be several DBA's, and some hierarchical structure may be desirable. The Applications Administrators are shown in the diagram to indicate the need for close coordination with the EA. As stated earlier, the AA's primary function is to support and represent the users' point of view. In this regard, the AA's probably should not be under the direct control of the EA, but rather, should belong to the various user organizations (hence the dotted line).

```
                    +-----------------------+
                    |Enterprise Administrator|
                    |        for the         |
                    |  Knowledge Resource    |
                    +-----------------------+
                                |
                                |
               +---------+      |
               |Knowledge|      |       +-----+
               |Resource |------+-------|Staff|
               |Center   |      |       +-----+
               +---------+      |
                                |
                                |
       +-----------+------------+-------+--------------+
       |           |            |       |     .        |
       |           |            |       |              |
  +----------+     |      +----------+  |     .  +----------+
  |Knowledge |     |      |Knowledge |  |        |Knowledge |
  |Resource  |     |      |Resource  |  |     .  |Resource  |
  |Policy    |     |      |Security &|  |        |Standards |
  +----------+     |      |Privacy   |  |     .  +----------+
                   |      +----------+  |
                   |                    |     .
       +-----------+-----+              |
       |                 |              |     .
       |                 |              |
  +-----------+   +----------+    +--------+   +-------------+
  |External   |   |Knowledge |    |Data    |   |Applications |
  |Information|   |Resource  |    |Base    |   |Administration|
  |Sources    |   |Technology|    |Admini- |   +-------------+
  +-----------+   |Assessment|    |stration|
                  +----------+    +--------+
```

KNOWLEDGE RESOURCE FUNCTIONS

Figure 6-7

The job of an EA is extremely complex and diverse. No clear-cut list of desired skills (political, managerial, or technical) has been developed at present. It is entirely possible that a department may need different mixes of skills and functions as they move from one phase of creating a knowledge resource to another. At one point, an organization may feel the need for an office of knowledge resource administration and at some other point they may not need a knowledge resource administrator. It seems clear, however, that the persons or organizations which will be responsible for creating the knowledge resource will need appropriate access to upper management. In any case, the exact placement of the EA function cannot be determined before the initial decision to create a knowledge resource is made. For instance, if mission knowledge is identified as the target of a knowledge resource program, then perhaps mission managers should organize and control the function which we have identified. If, however, knowledge relative to all of the disciplines of the organization is the target, then different mechanisms for organizing and controlling these functions would seem to be needed. Policy and organizational decisions such as these are the province of the Secretary and the management council of the department.

6.3 PHYSICAL VIEW

The Physical View of an enterprise not only reflects the current hardware/software environment of the organization, but also encompasses the state-of-the-art components which the enterprise needs to acquire or develop. The Physical View is, of course, most relevant to the tools and support context of the department. In many departments, the knowledge functions described in the Functional View might be performed on different computer systems (from a mix of vendors) with different file systems, file management systems, data base management systems, and knowledge base management software. In addition, several mass archival storage devices could be connected to individual computers or to some set of machines (e.g., IBM 3850 or AMPEX TBM). There have been few limitations on computer type, terminal type, software packages, or mass storage devices which a department may employ. The resultant environment may present formidable barriers to the creation of a knowledge resource from either the computer context or from the other contexts which employ the components of the Physical View. An orderly Physical View is necessary for the proper selection of hardware and software components to manage the knowledge resource optimally. A decision to choose a standard terminal subsystem, for example, would be a significant move toward an orderly Physical View. Figure 6-8 shows part of the potential configuration of a network with some of the variety of data management software supported by the machines.

The philosophy of treating knowledge as a departmental resource assumes that there is considerable need for sharing of data, information, and knowledge among the various divisions of the enterprise. Today's technology in computer networking (namely that employed on the ARPANET network) can provide the communications links for sharing data among several machines. However, this technology is severely lacking in providing an adequate vehicle for communicating the logical structures of and the relationships among the data involved, i.e., the factual knowledge. Things are even less developed in the area of procedural knowledge sharing. Strictly speaking, data sharing implies the need for similar processing capabilities at each node in the network, and hence, duplication of effort. The same data is available for processing by several different nodes, but the results of one processing activity are generally not available to another. The various systems merely process the same data in somewhat different fashions in order to derive different information pertinent to their own local activity.

```
        ........         .......         ..........
        :INGRES:         :M-204:         :DMS-1100:
        :......:         :.....:         :........:
            |                |                |
            |                |    +----+      |
......  +------+           +---+ /|3850|   +------+
:Prod:  |DEC   |  ......   |IBM|/ +----+   |UNIVAC|
: 3  :--|PDP-11| :IDMS:--  |370|           |1110  |
:....:  +------+ :.....    +---+           +------+
            |                |                  |
            |                |                  |
            |                |                  |
            |           ************            |
            |-----------*          *------------|
                        * COMPUTER *
                        *          *
                        * NETWORK  *
            |-----------*          *------------|
            |           ************            |
            |                |                  |
            |                |                  |
            |                |                  |
            |      +---+     |                  |
......  +------+ /|TBM|  +----+           +---------+
:DBMS:  |DEC   |/ +---+  |HIS |           |BURROUGHS|
: 10 :--|PDP-10|         |6060|           |  B6700  |
:....:  +------+         +----+           +---------+
            |                |                  |
            |                |                  |
        ........        ........            .......
        :System:        :IDS-II:            :DM 2 :
        :1022  :        :.......            :.....:
        :......:
```

HYPOTHETICAL HETEROGENEOUS COMPUTER NETWORK

THE PHYSICAL VIEW

Figure 6-8

The goal of a department should not be data sharing, but rather, knowledge sharing, where each process in the system performs some function on the data, derives some useful knowledge from it, and then makes this knowledge available to subsequent processes for additional knowledge production. Each step in the series not only puts the data to some local use, but adds value to the data by bringing together in new relationships some previously unassociated data items, thus enhancing the metadata base. In addition, new procedural knowledge might also be developed and made available. In this fashion, the value of the entire knowledge resource is enhanced for all members of the organization.

In the real world imposed by the Physical View, the sharing of knowledge is not an easy task. Great disparities exist among the hardware and software components, and the technology necessary to overcome these differences can indeed be complex. Of utmost importance (but, indeed, difficult) is imparting to humans or to other systems the representation of the logical structure of the data bases, i.e., the factual knowledge which they contain. Making procedural or judgmental knowledge available will be even more of a challenge. The automatic communication of the meaning of a data base or of a knowledge base appears to be beyond the state-of-the-art today, and yet this is precisely what is needed before we can truly claim to have a knowledge resource shared among a number of machines.

There are several approaches which we may take in attempting to alleviate some of the technical problems of data sharing in a computer network [31]. The applicability of these solutions to the broader problem of knowledge-sharing has not yet been explored. Four of these approaches are listed below.

1. Standardization - an organization may decree that identical hardware and/or software will be used throughout the network thereby removing the disparity among systems. Such an approach may be impractical or imprudent for a variety of reasons including economic, political, legal, or technical factors. The impact of standardization on existing equipment and programs in a department might be so enormous as to rule it out as a viable solution. An extensive cost/evaluation study would be required.

2. Centralization - a department may deem it appropriate to require that all knowledge processing and access be performed at a central site on a single or homogeneous set of machines intended solely for that function. The fact that centralization can imply a single point of failure for the network together with the problems of communication speeds for large volumes of data make centralization less than ideal. A modification of the approach is Distributed Centralization (e.g., via a back-end data management machine [32,33,34]) where identical hardware/software is distributed throughout the network to perform data management functions exclusively. This approach is intended to improve performance, increase reliability, and provide locality of data to the processes which use it.

3. Transformation - a third alternative is the automatic transformation of discrete data structures from one system to another throughout the network. Assuming such a transformation (including subsequent communications) could be performed rapidly enough to be of use, the resultant duplication of data and the inherent problems of keeping duplicate data bases in synchrony make transformation an unlikely solution for knowledge sharing. Yet, there is still the very real problem of system migration from one generation of hardware or software to another, and transformation technology can greatly assist in this costly and time-consuming process.

4. Translation - the fourth alternative for a department is to construct a translation mechanism whereby requests for information in one system can be automatically translated into requests comprehensible to other systems as appropriate. The translation mechanism may be either centralized at one local site or it may be distributed throughout the network, possibly in the form of front-end machines. In either case, an important theoretical problem still exists concerning whether or not it is even possible to translate complex requests for information into terms which a simple-structured data management system can understand. In addition to this, there is also the problem of the unrestrained propagation of data management systems and the constant need for additional translators throughout the life of the network.

Distributed centralization (back-ends) and distributed translation (front-ends) appear to be the two most technologically sound approaches. Their difference is one of emphasis: whether all data management can and should be performed by a single kind of system so that requests for knowledge from various nodes in the network need not be translated, or whether the advantages of each unique data management system are sufficient to warrant their retention at the cost of continually translating requests and responses. Other approaches are undoubtedly possible, including hybrids of the four mentioned above. Each department will have to decide the approach most appropriate to its own unique set of circumstances.

Regardless of the approach taken by a department, there remain many other problems in attempting to share knowledge in a computer network. There are problems of distribution, duplication, and synchronization among data bases. There are problems of concurrent updating, network locking strategies, and multiprocessor deadlocks. There are problems of locating all pertinent data from a number of sources, data base navigation through unknown structures, and the need for a network-oriented dictionary/directory. Finally, there is the unique security problem of the vulnerability and sensitivity of the network directory and the navigational facility. Therein lies the key to the organization of knowledge of the entire department. Prior to the establishment of such a facility, the knowledge relevant to data base organization was distributed throughout the various processing centers. Access to one did not provide access to or

knowledge about the functioning of any other. Security was accomplished by obfuscation. With this facility, everything which the department does with data is described in a neat and orderly fashion. Such prized knowledge makes the network directory and its associated features a prime target for attack.

6.4 RELATIONSHIP OF THE THREE VIEWS

Given that the Functional, Structural, and Physical Views are all views of a knowledge resource within an enterprise, it is interesting to determine how they correspond to each other and where they overlap. The Physical View considers the actual hardware and software employed by a department to carry out the functions necessary to its mission. This View identifies the actual file systems, file management systems, and data base management systems of the Functional View which are used to store and maintain the data throughout the department. The Physical View lists the tools which are available to the various people identified in the Structural View to enable them to do their jobs.

The Structural View is related to the Functional View as follows. The conceptual model is concerned with overall organizational productivity and the flow of data from process to process (node to node) as it moves about in the system (where the system may be a single computer or an entire network). Consequently, the conceptual model addresses the areas of interaction among processes. The internal model deals with the efficient performance and maintenance of individual processes within a node. Hence, the internal model gives a microscopic look at the nodes of the system. The external model is the user's view of the system and hence is concerned with the interfaces (access languages, application programs, etc.) which access the various nodes.

From the external point of view, the users would like to see a uniform front, i.e., standard interfaces. They want interfaces which facilitate their working with a particular knowledge base regardless of what it is. They do not, nor should they, care about the underlying machinations required to supply them with the data they need to support their knowledge applications. While the users are aware that requests for service cost money in terms of hardware and software resources expended, it matters little to them whether the request is translated or transformed or whether it is processed locally or remotely. The main costs which the users see are the human costs involved in their use of the system. Users tend to get very upset, and rightfully so, with systems where trade-offs in cost have been made in favor of hardware and software at the expense of increased human costs and increased difficulty in developing their knowledge applications. Inadequate human/machine interfaces have been provided in the past which have been designed to favor machine efficiency over the efficiency of the human users. The things that are important to the users are the validity of the response and the basis for it, the timeliness of the data, the speed with which the request is processed, and the ease with which they can obtain what they

need from the system. In addition, the system must be able to accommodate the changing mission environments with which the users must deal. Users know, only too well, that they are working on problems which keep changing. Computerized systems must be flexible enough to adjust to the continual changing requirements and specifications which are common and necessary. In seeking to develop a knowledge resource an organization must recognize these important aspects and organize their system accordingly. Failure to do so will result in the overall failure of a department's effort to optimize the use of all of its resources.

From the internal point of view, a Data Base Administrator would like to see a consistent and efficient system. Standardization of the system can make the job easier in terms of training and maintenance efforts, but standardization can also have a negative effect on performance. The DBA is caught in the unenviable position of having to provide adequate service to the users while adhering to enterprise policies based on global optimization.

From the conceptual point of view, an Enterprise Administrator should be worried about making the entire thing work together for the benefit of the department as a whole. An EA might need to step on a few toes in the struggle, and he or she might have to pursuade the users to exercise a proper amount of restraint to ensure that global priorities are met. The EA should be responsible for coordinating the various DBA's to ensure that ample facilities are present to allow knowledge sharing to take place.

The three Views (Functional, Structural,and Physical) provide three ways to view a knowledge resource in a large complex enterprise. They are each related to the others in that they are modeling the same process, but they differ in their emphasis and point of view. Each View provides a useful way of looking at certain subsets of the overall system depending on the interest of the observer.

7. STEPS TOWARD REALIZING A DEPARTMENTAL KNOWLEDGE RESOURCE

Once a department reaches the conclusion that knowledge is, indeed, a valuable resource and one which should be created and developed, what could a department do to allow this to come about? There are many problems, both technical and managerial, which an organization should expect to face in moving from their current position to one where knowledge is a resource. In the following sections we present some technical and managerial issues which will require solutions before an organization can create an effective knowledge resource.

7.1 TOP MANAGEMENT COMMITMENT

First on the list of things to do is to obtain the written and public commitment of top management to explore the knowledge-as-a-resource philosophy. As we stressed earlier, failure to obtain such a firm commitment at the outset so undermines the concept as to almost guarantee the failure of an implementation. Before a knowledge resource policy can be implemented management must be convinced of the appropriateness and desirability of recognizing knowledge as an organizational resource. This is not a small task to be brushed over hurriedly in a one-hour briefing. Top management must understand the concept of a knowledge resource and they must be convinced that it is both viable and beneficial, i.e., that in the final analysis it will save money and/or increase capabilities. This commitment should be preceded by a study which includes a statement of the organization's goals and objectives and an assessment of the current state of affairs in data management and knowledge-based systems technology. In addition, some preliminary plans and cost analysis for the next several phases of the implementation will be required before management can make an informed decision. The preliminary plan must be flexible enough to allow organizational change as the organization proceeds toward the creation of an effective knowledge resource. The plan, in effect, could serve as a basic knowledge resource charter for the department by outlining what is to be accomplished without specifying how it is to be done. This latter point is most important since the impact of this philosophy will ripple throughout the department. Freezing the steps to a knowledge resource too early may force a pace that is either too rapid or not fast enough. Organizational learning must be built into the plan, and its pace will be difficult to predict accurately (at least in the early stages).

7.2 PUBLIC RELATIONS

Once top management has been convinced of the important potential of viewing knowledge as a departmental resource, the job remains for someone to spread the word among the various organizations within the department. A base of support in the entire department should be built by informing pertinent personnel of the goals of the new approach and by soliciting their assistance. During this phase the groundwork is being laid for unification and smoother working relationships in the years

ahead.

7.3 MODELS OF ACTUAL CONDITIONS

If a department decides that an EA and associated staff are needed, then they can be assigned the responsibility for developing the knowledge resource. Otherwise, the responsibility will need to be assigned to some existing organizations. In any case, one of the first things to be done is to identify and classify as much of the organization's knowledge (all types) as possible. This will help the department understand what it is that will be in the resource. Another thing to be done is to build a set of models of the Functional, Physical, and Structural Views of the department (see Section 6). These views should not be limited to a description of the current state of affairs but should also include an assessment of where the department could be, should be, and probably will be in the next 5 years. These models are very important to the organization's knowledge resource planning, for they carry in them all the work that needs to be done if the department is to move from its current state to one wherein knowledge can be an effective resource of the department.

7.4 JOINT FUNDING OF RESEARCH

A department should not try to "go it alone" in building a knowledge resource. It is doubtful that any single enterprise has sufficient money and expertise to successfully complete this endeavor alone. Instead, they should seek out relevant committees to join and support such as those of ANSI, CODASYL (Committee on Data System Languages), and the GUIDE and SHARE groups of users of IBM equipment. A department needs to participate in these organizations both to gain knowledge and to have some impact on the direction of these efforts. Key individuals and other organizations with similar problems should be identified and contacts established so that experiences can be shared and, where possible, resources can be pooled. Conferences and meetings are an excellent source of such contacts. Also various research efforts should be identified (see Section 7.5) and classified as to whether they should be funded by the department or merely tracked. The lucrative possibilities of joint funding of research and development in areas of mutual interest must not be overlooked.

7.5 KEY RESEARCH TOPICS AND PROBLEM AREAS

In Appendix-1 we outline some of the key issues which are relevant to an organization's decision to treat knowledge as a resource and point out some of the problems which an organization should expect to face. This appendix also lists several research areas where tools and theory need to be developed in order to treat knowledge effectively as a corporate resource. Perhaps the key item in the list is the development of a Knowledge Resource Center for the organization, a focal point where all aspects of knowledge production can be monitored and directed. Essential to the functioning of such a center is the collection of knowledge

about data bases under the control of the enterprise, the maintenance of a dictionary and directory of data items in any of the data bases, a knowledge base management system for maintaining the center's own data bases, and a methodology for representing to management the knowledge and data which the center contains. The purpose of the Knowledge Resource Center is to provide immediate answers to questions dealing with the availability and general location of pertinent knowledge, not to answer specific low-level factual questions. It is intended to give top-level management a handle on the whole process of knowledge-production within the department.

Other areas have need for research in knowledge augmentation. Models of the organization corresponding to the Functional, Structural, or Physical Views as well as the conceptual, internal, and external schemas are useful if a department is to have a proper understanding of the role and importance of each. Knowledge-based approaches are suggested to enhance the development of workable, human-oriented interfaces to the many aspects of a knowledge resource within the organization, e.g., natural language, forms, or report production. Related to this concept is the need for knowledge-directed navigational facilities to assist knowledge resource users as they wind their way from data base to data base searching for knowledge pertinent to their tasks. In the area of converting data into knowledge lies the need for a methodology for applying validity values to individual data items, to relationships among data items, and to entire data bases according to their source. This validity must be computationally usable in order to arrive at validities for compound situations. In addition, there is the need for assistance in correlating data from a variety of sources. Next there is the need for semantic consistency checks of data to be entered into a data base, i.e., filters on impossible and non-meaningful data. Finally, there is the need for knowledge-based security systems which monitor data base activity, which have an understanding of what the data is about, and which can make inferences about what potential security violations might arise by granting access for a given user to a particular piece of data at a specific point in time.

Each of these topics, along with many others, is discussed in greater detail in Appendix-1. For convenience, the topics have been ordered with regard to the Structural View with the research areas being grouped under the functions which were identified in Section 6.

8. LOGICAL SYSTEM DESIGN

In this section we present a logical system design which might serve as a basis for implementing a knowledge resource in a department. The details of the implementation are, of course, not yet firm, but the overall model is one which we think provides both the tools and flexibility which will be needed if knowledge is to become a departmental resource. Additional thought and design will be required before a full-scale implementation is tried, and each department may want to take the general design and create their own specific implementation. Regardless of the specifics, we believe that the general components which are identified in the design will need to be included in most implementations. The architecture of a logical system design which is to be used to develop a knowledge resource must be able to survive the inevitable changes in direction and technology which are to be expected with an activity of this nature. Moving an organization from the world of data processing to a world where knowledge is a departmental resource must necessarily be a learning process for all involved. Undoubtedly, the organization's perception of knowledge and of what constitutes a knowledge resource will grow and change with experience.

Several aspects of the design are crucial to its success. The following list of general characteristics is indicative of the dynamic environment which any proposed logical system design must be able to support.

* Evolutionary - the system design must be evolutionary to allow for an orderly transition from the current structural, functional, and physical environment of the department.

* Extendable - the system must be extendable from the initial design so that new knowledge resource tools can be added or deleted with minimal impact.

* Modular - the system must be implemented in a modular fashion to allow easy construction, easy maintenance, and a failsoft capability.

* Incrementally Usable - the system must be incrementally usable so that benefits can begin to be received prior to the completion of the entire system.

* Powerful - the system must be powerful enough to support all of the ramifications of having a knowledge resource as indicated in Sections 2,3,and 4.

* Cost/Risk - the system design must make maximum use of existing technology both to keep the cost within reason and to minimize the risk involved with embarking on new research efforts.

* Enterprise View - the architecture of the logical system design must support the enterprise point of view above individual parochial interests.

* Learning Process - the system must be able to incorporate the learning process which the organization will have to go through in order to provide a knowledge resource.

Above all else, the system must be manageable. It must allow for numerous points of managerial interaction and control by the humans it is to serve. The system is intended to provide a set of tools to assist departmental personnel in accessing their knowledge, not to replace them.

In addition to the above-mentioned general characteristics, there are several aspects of a knowledge resource which the design specifically addresses. These issues stem from the preceding discussions on knowledge independence, the need for multiple interfaces and data managers, and a knowledge-based approach. It is our opinion that these aspects are crucial to any system design for a knowledge resource. The logical system design of this section is one attempt at embodying these issues.

(1) Requirements for any proposed system must be gathered with a clear understanding of the knowledge components.

(2) Knowledge independence must play a key role in the design.

(3) One aspect of knowledge independence is the separation of user oriented views of knowledge from the computer oriented views.

(4) Multiple user interfaces are required to provide the appropriate view of knowledge which is most natural to a particular user category.

(5) Multiple knowledge resource tools are needed to provide adequate capabilities to meet differing knowledge requirements.

(6) A canonical representation of knowledge will be required for knowledge sharing among distributed systems.

(7) Some facility will be required to assist in the navigation of distributed factual and procedural knowledge bases.

We hope that the logical system design outlined below will make these issues more explicit for the reader.

In Section 6.2 (the Structural View) we described three models of information as postulated by the ANSI/SPARC group: the internal, the external, and the conceptual models. In an earlier paper [12], we identified a logical system design which corresponded very closely to the ANSI model. For each component in the ANSI model there was a component in the logical system design: a Data Management Subsystem (DMS), a User Interface Subsystem (UIS), and a Translation and Control Subsystem (TCS). Figure 8-1 shows these three subsystems.

```
+----------+      +-----------+      +---------+
|Data      |      |Translation|      |User     |
|Management|------|and Control|------|Interface|
|Subsystem |      |Subsystem  |      |Subsystem|
+----------+      +-----------+      +---------+
      |                 |                 |
      |                 |                 |
   /-----\         +-----------+         /\
  |       |        |Enterprise |        /  \
  |\_____/|        |Knowledge  |       /    \
  |       |        |Resource   |      /      \
  | Data  |        +-----------+     ( Users  )
  | Bases |                           '-------'
   \_____/


   \                /         \                /
    \              /           \              /
     \            /             \            /
      DATA BASE                  APPLICATIONS
     ADMINISTRATOR               ADMINISTRATOR

              \                    /
               \                  /
                \                /
                   ENTERPRISE
                  ADMINISTRATOR
```

LOGICAL SYSTEM DESIGN

Figure 8-1

If an organization desires to focus on the factual knowledge component of a knowledge resource then this early design might be satisfactory. However, it does not provide separation between all of the types of knowledge since procedural and judgmental knowledge are not identified as distinct subsystems. Therefore, we have extended and modified the logical system design to provide for separation of procedural and judgmental knowledge. The extended model contains three subsystems which correspond to each type of knowledge in the taxonomy: Factual Knowledge Subsystem (FKS), Procedural Knowledge Subsystem (PKS), and Judgment Support Subsystem (JSS). In addition, we have retained the Translation and Control Subsystem (TCS) to fill the continuing need for integration between the subsystems of the logical system design. This extended design is shown in Figure 8-2. The breakout of the components of the various subsystems is shown with their respective subsystem in subsequent figures. Provision for a network environment such as the one presented in the Physical View of Section 6.3 is contained in the Translation and Control Subsystem and appears in Figure 8-5. The actual software packages selected by a department to comprise its Physical View are not important to the understanding of the logical system design.

```
                            (Enterprise
                            Administrator)

     FACTUAL                                            JUDGMENTAL
     KNOWLEDGE                                          KNOWLEDGE
     BASES                     to
                             Enterprise
     (Data Base              Knowledge                 (Applications
     Administrators)         Resource                  Administrators)
                             Center

    +----------+               / \
    |Factual   |                |
    |Knowledge |--+             |
    |Subsystem |  |             |
    +----------+  |        +-----------+         +---------+
                  |        |Translation|         |Judgment |
                  +--------|and Control|---------|Support  |
                  |        |Subsystem  |         |Subsystem|
    +----------+  |        +-----------+         +---------+
    |Procedural|  |             |
    |Knowledge |--+             |
    |Subsystem |                |
    +----------+               \ /

                               to
     PROCEDURAL               other
     KNOWLEDGE              knowledge
     BASES                   systems

     (Procedure
      Administrators)


     INTERNAL               CONCEPTUAL               EXTERNAL
       VIEW                    VIEW                    VIEW

    (Computers)             (Enterprise)              (Users)
```

SYSTEM ARCHITECTURE FOR A KNOWLEDGE RESOURCE

Figure 8-2

8.1 FACTUAL KNOWLEDGE SUBSYSTEM

The Factual Knowledge Subsystem (FKS) is the purview of the Data Base Administrator. The FKS provides an internal view of the actual organization of data and makes available a variety of data management software (access engines) as described in the Functional View of Section 6. We suggest that as a minimum an operating system-supplied file system, a file management system, and several data base management systems (DBMS) be included. The operating system-supplied file structure will allow the users to create and keep their own personal files which may serve as a personalized part of the knowledge resource. A text editor user interface operating on standard files will provide the capability to handle the large volume of written reports and correspondence which are typical of today's modern organizations. This facility will also allow this type of valuable knowledge to be incorporated into the department's knowledge resource. The file management system provides standardization of the management of files where data volumes are high, where limited logical views of data are needed, and where cost is the overwhelming consideration. Several types of DBMS are available for the DBA to choose from including inverted files, hash-coded, plex structure, and relational systems. A plex structure DBMS will allow the organization to build complex network views for data whose structure is relatively stable. A relational DBMS can be used to provide rapid access to data whose structure is fairly dynamic. Depending on the manner of implementation, an inverted file system may be considered to be an advanced file management system or as a special case of a relational system where rapid retrieval is required from a data base with a relatively stable structure and a limited volume of updates. Hash-coded systems offer rapid access to data by specific, unique keys. Each type of DBMS has its advantages and disadvantages. None is suitable for all users. Consequently, a selection of factual knowledge managers should be available if a DBA is to choose the data manager most appropriate for a given task. Figure 8-3 shows the Factual Knowledge Subsystem and its generic components.

```
           +----------+
relational,|Data Base | +-------------+
plex,      |Management|-|DBMS-specific|-|
hash-coded,|Systems   | |Interfaces   | |
           +----------+ +-------------+ |
inverted                                |
file,                                   |
           +----------+                 | +-----------------+
file       |File      | +-------------+ | |Factual Knowledge| to
management |Management|-|FMS-specific |-|-|Subsystem        |--->
system,    |System    | |Interface    | | |Translator       | TCS
           +----------+ +-------------+ | +-----------------+
                                        |
                                        |
           +----------+                 |
files      |Standard  | +-------------+ |
           |File      |-|SFS-specific |-|
           |System    | |Interface    |
           +----------+ +-------------+


              ACCESS     ENGINE-SPECIFIC       GENERAL
              ENGINES      INTERFACES         INTERFACE
```

FACTUAL KNOWLEDGE SUBSYSTEM

Figure 8-3

8.2 PROCEDURAL KNOWLEDGE SUBSYSTEM

The Procedural Knowledge Subsystem (PKS) is tasked with managing the procedural knowledge which will be available in the knowledge resource. The PKS is divided into algorithmic and heuristic components. The former contains the conventional computer programs which will perform most of the routine processing in the knowledge resource and includes such things as statistical routines, mathematical routines, and programmed algorithms of all types. It also includes all of the facilities which can be used to support the development of algorithms which will need to be added to the knowledge resource.

The heuristic component of the PKS will contain systems such as the knowledge-based systems which were discussed in Section 5.1. Ultimately, it is expected that there will need to be a rich variety of knowledge-based systems available for use in the knowledge resource. The PKS depicted in Figure 8-4 will be managed by people called Procedure Administrators who will be responsible for the difficult task of keeping track of the large numbers of algorithmic programs. In addition, they will provide technical assistance in the use of the individual knowledge-based systems which are provided in this subsystem.

```
+-----------------------+
| +---------------+     |
| |               |     |
| |  Algorithm  1 |-+   |
| |               | |   |
| +---------------+ |   |
|   | Algorithm  2  |   +--+
|   |               |-+ |  |
|   +---------------+ | |  |
|     | Algorithm m   | |  |
|     |               | |  |
|     +---------------+ |  |
+-----------------------+  |
                           |
                           |
                           |  +----------+
                           |  |Procedural| to
                           +--|Knowledge |--->
                           |  |Subsystem | TCS
                           |  |Translator|
                           |  +----------+
+-----------------------+  |
|                       |  |
| +---------------+     |  |
| |  Heuristic    |     |  |
| |  Procedure  1 |-+   |  |
| |               | |   |  |
| +---------------+ |   |  |
|   |  Heuristic    |-+ |--+
|   |  Procedure  2 | | |
|   +---------------+ | |
|     |  Heuristic    | |
|     |  Procedure  n | |
|     +---------------+ |
+-----------------------+
```

PROCEDURAL KNOWLEDGE SUBSYSTEM

Figure 8-4

```
                            USER
                            /\
                            \/
                      +---------------+
                      |   Knowledge-  |
                      | Based Personal|
                      |   Assistant   |
                      +---------------+
                              |
                      +-------+-------+
 +--------+           |       |       |
 |Natural |           |  +---------+  |          +--------+
 |Language|--|        |  |Utilities|  |          |Query   |
 |Subset  |  |        |  +---------+  |       |--|Language|
 +--------+  |        |               |       |  +--------+
             |        |               |       |
+---------+  |        |               |       |  +----------+
|Graphical|--|        |               |       |--|Relational|
+---------+  |        |               |       |  +----------+
             |  +-----------+    +---------+  |
    +-----+  |  |Structure  |    |Structure|  |  +------------+
    |Forms|--|--|Independent|    |Dependent|--|--|Programmatic|
    +-----+  |  |Interface  |    |Interface|  |  +------------+
             |  +-----------+    +---------+  |
    +-----+  |        |               |       |  +------------+
    |Menus|--|        |               |       |--|Navigational|
    +-----+  |        |---------------|       |  +------------+
             |                |               |
+---------+  |                |               |  +-----------+
|Dialogues|--|        +----------------+      |--|Text Editor|
+---------+  |        |Judgment Support|         +-----------+
             |        |    Subsystem   |
  +-------+  |        |    Translator  |
  |Trans- |--|        +----------------+
  |actions|                     |
  +-------+                     |
                             \ /
                            to TCS
```

Data Structure
Independent
Interfaces

Data Structure
Dependent
Interfaces

JUDGMENT SUPPORT SUBSYSTEM

Figure 8-5

8.3 JUDGMENT SUPPORT SUBSYSTEM

The Judgment Support Subsystem (JSS) is the mechanism through which most of the human users will access the knowledge resource. It also is the component where these humans will be provided the opportunity to exercise their judgmental knowledge. Because the JSS is intended to serve as the basic mode of access to the knowledge resource, it has been designed to contain a wide variety of interfaces which are independent of the other subsystems and which are also independent of each other. Providing such a variety of capabilities serves the dual purposes of maintaining maximum knowledge independence and supporting user-oriented views of the knowledge resource. The JSS is shown in Figure 8-5.

The users in an organization will be faced with a complex environment as more and more of the components of a knowledge resource are implemented and connected. We believe that the system itself can be constructed in a fashion to supply the users with help in navigating and manipulating this environment. We see this role being filled by something which we refer to as a "knowledge-based personal assistant" (KBPA) which employs the knowledge-based systems technology as described in Section 5. The KBPA will be tuned to a particular user and will contain substantial amounts of knowledge about what that user needs to do his or her job. It will need to know which types of factual knowledge the user employs in his or her work. It will need to know about the different procedures which a given user needs to access, and it will contain rules on the mission skills and targets which form the user's domain. In short, it is hoped that KBPA's will be able to function as reasonably intelligent assistants to help the human user function in the knowledge resource environment. This proposed KBPA is not as far-fetched as it might seem. Knowledge-based systems such as R11A [25] have already been constructed to perform some of the functions which we have just described. Increases in capabilities are needed, of course; however, trends in the intelligent terminal area indicate that a "knowledge-based personal assistant" may be technically possible within the next few years.

The users of a knowledge resource will need access to a variety of user interfaces which should be standardized throughout the department. These interfaces should be selected to provide the broad range of capabilities which will be needed to support a user population whose requirements cover a full spectrum. We feel that no single user interface will be capable of satisfying all users' needs. Instead, the key to successful user interface modules is to provide many standardized interfaces from which a user or group of users can select those most suitable to their immediate needs. These interfaces may require different levels of skill to operate, but each, in turn, could provide significant differences in terms of flexibility and capability. The issue of importance is not whether one interface is superior to another, but rather whether users can be provided with a set of consistent and stable interfaces which are

supported across all systems in their department. The Judgment Support Subsystem in Figure 8-5 identifies 11 different user interfaces. Undoubtedly, not all user interfaces will be required by all systems, and some useful interfaces may have been omitted from this list.

The user interfaces can be grouped into two categories depending on the relative degree of independence that the interface has from the underlying data structures and data managers that it accesses. Those interfaces having a relatively high degree of independence include: (1) a subset of natural language (English) which can be used in a limited-context environment to allow queries to be expressed in a more-or-less natural language form, (2) a graphical interface which provides information-rich pictorial representations of relatively large amounts of data (such as charts, graphs, diagrams, etc.), (3) a forms-oriented interface to allow both data entry and retrieval via forms in an attempt to simulate and replace the existing world of paper forms prevalent in most organizations, (4) a menu-processor which allows the selection of functions from a list presented dynamically to the user, (5) a set of dialogues which permit interaction with specific levels of users in a vocabulary and format which is commensurate with the user's position and function, and (6) a transaction-oriented interface which allows rapid response to terse commands which represent individual, independent actions to be taken.

Those interfaces which require a relatively high degree of dependence on the specific underlying data manager (i.e., low independence) include (7) a query language (of which there may be several); it is hoped that the need for a separate query language will eventually be obviated by the natural language subset interface, (8) a relational interface which allows greater flexibility in query expression, but which currently requires a relational view of data, (9) one or more programmatic interfaces to specific programming languages for use by application programs which are intended to perform calculations based on the data being accessed, (10) a navigational interface which permits the user to browse or navigate on two levels: (a) when tightly connected to a structured data base management system, and (b) when accessing the knowledge resource description contained in the Translation and Control Subsystem, and (11) a text editor interface which provides the capability of interactively entering and editing data and textual material which is stored in standard files.

Two final points about the user interfaces remain to be considered. First, a set of common user utilities such as sorting, report generation, etc. is required and needs to be accessible from each user interface. By placing these utilities close to the user interface, problems such as different collating sequences are made easier. Second, another layer of software, in addition to the KBPA, may become necessary between the users and the various user interfaces. Its function is to provide standardized user interfaces and protocols which have command languages for selecting the various user interfaces, moving from

one to another, and calling on the utilities. A "universal" interface perhaps is enterprise-dependent but may also be user-dependent in the sense that whole classes of users may require a common interface unique from other classes.

8.4 TRANSLATION and CONTROL SUBSYSTEM

The Translation and Control Subsystem (TCS) is the mechanism which integrates all of the other subsystems of the logical systems design. Figure 8-6 shows some of the modules which will be required by the Translation and Control Subsystem. Central to the translation process is a canonical form for data structures and data formats which is sufficiently general to accommodate the representation of any data structure employed by a data manager, any form of representation employed by a user, and any data format employed by a computer with which it must communicate. It must allow, too, for direct linkages between data structure dependent user interfaces and the appropriate data managers without throwing up a roadblock of unnecessary translation. It is important to keep clear the difference between the canonical representation and the mapping mechanism necessary between this canonical form and the other modules of the system. This mapping process is termed translation, while the canonical form is the intermediate representation through which the translation is performed.

The TCS does more, however, than just translate between the judgmental support subsystem and the procedural and factual knowledge subsystems (no trivial task in itself). The TCS incorporates the ability to access all types of knowledge throughout the entire knowledge resource. Thus, the Translation and Control Subsystem must contain a data dictionary and data directory consistent with the entire department as well as data location algorithms to send requests to the appropriate data managers wherever they may be located (locally or in a computer network -- see Figure 8-7). The TCS must provide the capability to the users to access one or more local data bases, one or more remote data bases, or some combination of these. Presumably, in the early stages of development explicit directions from the user will be required to determine the particular data bases to be accessed. In more advanced stages, the system may be expected to determine the range of distribution required from the context of the user. In this capacity the dictionary/directory plays a key role and must be kept reasonably accurate. It does not, however, have to contain all data about data; a mere outline or synopsis may suffice to direct the search to potentially relevant files and data bases.

The Translation and Control Subsystem also is the natural place to establish a connection to the department's knowledge resource via a computer network (see, again, Figure 8-7). The canonical representation provides a common facility for sharing knowledge among disparate systems while also feeding data about the knowledge bases themselves into some centralized enterprise-level control mechanism so that an overall organizational view

can be maintained. In this capacity a TCS knowledge-based system (TCS/KBS) will be required to allow such activities as inductive and deductive inferencing with rule sets relevant both to the local system environment and the enterprise knowledge resource as well. Because the Translation and Control Subsystem is expected to employ canonical forms for data and procedural knowledge, the amount of effort needed to feed data and knowledge into the enterprise knowledge resource should result in minimal overhead. That is, the features for a knowledge resource center are built-in, not added-on.

One final function of the Translation and Control Subsystem is to exercise control over the transition between data managers and user interfaces. In this capacity, the TCS must have some data management capabilities of its own, must maintain some data bases relevant to its special function, and must coordinate matters of security, integrity, recovery, authorization, and concurrent access. These problems are especially difficult in a distributed environment where different data managers and even different computers may be involved. Considerable research in these areas will be needed before a final working system can be developed.

```
          +-------------+ +------------+ +--------------+
          |Distributed  | |Distributed | |Distributed   |
          |Dictionary/  | |Factual Know-| |Procedural Know-|
          |Directory    | |ledge Locator| |ledge Locator |
          +-------------+ +------------+ +--------------+
                |              |               |
                +--------------+---------------+
                               |
                               |
     +-------------+           |
to   |Factual Know-|           |
<----|ledge Subsystem|--+      |
FKS  |Translator   |  |  +----------+  +---------------+
     +-------------+  |  |          |  |Judgment Support| to
                      +--|  Control |--|Subsystem      |--->
    +--------------+  |  |          |  |Translator     | JSS
to  |Procedural Know-| |  +----------+  +---------------+
<---|ledge Subsystem |--+      |
PKS |Translator     |           |
    +--------------+           |
                               |
                 +-------------+---------------+
                 |             |               |
          +-------------+ +-----------+ +----------------+
          |Canonical    | |TCS        | |Canonical Pro-   |
          |Data         | |Knowledge- | |cedural Knowledge|
          |Representation| |Based System| |Representation |
          +-------------+ +-----------+ +----------------+
```

TRANSLATION and CONTROL SUBSYSTEM

Figure 8-6

```
              +-----------+
              |Procedural |
              |Knowledge  |
              |Subsystem  |
              +-----------+
                    |
+----------+  +-----------+  +---------+ --
|Factual   |  |Translation|  |Judgment |    \
|Knowledge |--|and Control|--|Support  |     \
|Subsystem |  |Subsystem  |  |Subsystem|      \
+----------+  +-----------+  +---------+       \
     |              |            |              \
     |              |            |               > Distributed
  /-----\        +-------+      /\              /  System #1
 |       |       |Network|     /  \            /
 |\_____/|       |Control|    /    \          /
 |       |       |Program|   /      \        /
 | Data  |       +-------+  (  Users )      /
 | Bases |          |       `--------' --
  \_____/           |
                    |
              ************  +-------+  +-----------+  +----------+
              *          *  |Network|  |Translation|  |Enterprise|
              * Computer *--|Control|--|and Control|--|Knowledge |
              * Network  *  |Program|  |Subsystem  |  |Resource  |
              *          *  +-------+  +-----------+  |Center    |
              ************                            +----------+
                    |
  /-----\           |
 |       |          |            /\     --
 |\_____/|       +-------+      /  \      \
 |       |       |Network|     /    \      \
 | Data  |       |Control|    /      \      \
 | Bases |       |Program|   (  Users )      \
  \_____/        +-------+   `--------'       \
     |              |            |             > Distributed
     |              |            |            /  System #n
+----------+  +-----------+  +---------+     /
|Factual   |  |Translation|  |Judgment |    /
|Knowledge |--|and Control|--|Support  |   /
|Subsystem |  |Subsystem  |  |Subsystem|  /
+----------+  +-----------+  +---------+ --
                    |
              +-----------+
              |Procedural |
              |Knowledge  |
              |Subsystem  |
              +-----------+
```

DISTRIBUTED SYSTEM ARCHITECTURE

Figure 8-7

It should be pointed out that it is not necessary that every physical implementation of this logical system contain every module (factual or procedural knowledge component or every component in the judgment support subsystem). What we are advocating is that the various modules be available for system implementors to choose from according to their specific requirements. We emphasize the need to standardize on the various user interfaces so that users may transfer among systems throughout a department and effectively see no difference in the command languages or data presentation no matter which hardware they use.

The logical system design which we have described addresses to some extent the specific knowledge issues outlined at the beginning of this section. The separation of factual knowledge from procedural knowledge is addressed by the separation of the factual knowledge subsystem from the procedural knowledge subsystem as well as by the distributed dictionary and directory and distributed factual and procedural knowledge locators of the Translation and Control Subsystem. Multiple user interfaces are accounted for as are multiple data managers for more natural views of knowledge. Canonical representations of factual and procedural knowledge are addressed in the TCS. Finally, a facility for navigating the knowledge resource is included in the navigational user interface which can be used to access the directory and the factual and procedural knowledge locator modules.

Two additional issues remain to be discussed: data acquisition and archival storage. Data acquisition is really just a process which may employ one or more of the user interfaces listed above (e.g., programmatic, transaction, or forms). Departmental standardization on the data entry function can go a long way toward easing the problem of transferring personnel among various data entry applications.

The question of archival storage interface is left open. Several alternatives are possible, depending on requirements and resources. A mass storage device may be appended as a back-end to the file system in the logical design (for example, as in the IBM 3850); or it may be included as an additional data manager (employing translation via the TCS); or it may be implemented as a stand-alone centralized system accessible through the translation and control subsystem via the network (as with the Datacomputer). In any event, care must be taken to insure that knowledge placed in archival storage will always be retrievable. By this we mean that factual knowledge stored in a form appropriate for a particular data manager may be unusable in future years should that data manager be abandoned or replaced by the department. Some means must be provided for storing all data via a single mechanism, and that mechanism must be retained for the life of the archival storage. The canonical representation which are to be found in the TCS may be the appropriate vehicle for such long-term storage and access.

No claims are made at this time about the size of machine needed to implement this system. The fact that only parts of the entire design may be desired in a given installation indicates that some so-called minicomputers may be appropriate. Furthermore, the logical system structure described above need not be implemented on a single machine. In fact, the various subsystems may each be implemented on a different computer for modularity, expandability, performance, cost, or a combination of reasons. Use of the various interfaces and data managers need not wait for the development of the translation and control subsystem, nor must one package be dependent on the development of all the others. Direct linkages of user interfaces to data managers can be implemented as the modules are developed, but with the proviso that such direct linkages can be broken easily when the Translation and Control Subsystem becomes available. Each module can be developed individually (although with a coordinated goal in mind), and the system can be implemented in phases according to available resources and technological advances. Once the various subsystems have been stabilized, many of the functions may be moved to firmware for greater processing efficiency. Parallel processing may also be employed in any subsystem to further improve the response to the user.

8.5 UNIX PROTOTYPE

Some preliminary work has begun toward testing the feasibility of the logical system design described above using the UNIX operating system [35] on Digital Equipment Corporation PDP-11 computers. A discussion of the particular ideas and software products involved is contained in Appendix-2.

9. CONCLUSIONS AND RECOMMENDATIONS

A major premise of this paper is that most Federal departments are "knowledge agencies." By this we mean that their primary product is knowledge, be it factual knowledge about data related to the departmental mission, procedural knowledge about the techniques employed to process that information, or the judgmental knowledge by which policy and decisions are made. This total knowledge is central to the existence of a Federal department and should be formally recognized as the basic resource that it is. As we said in Section 1, the orientation of the authors has been primarily toward governmental activity, and the recommendations which follow reflect that orientation. The application of these remarks to other types of organizations is, perhaps, the subject of another paper.

9.1 KNOWLEDGE IS A BASIC RESOURCE

The formal recognition of knowledge as a basic departmental resource will have widespread ramifications as indicated in this paper. The most visible, perhaps, might be the recognition of the new functions which will be needed to maintain the organization's knowledge resource and to insure compatibility among all of the various groups and projects. Consistent with this philosophy is the need for line and project managers to reorganize their approach to knowledge applications and systems design. Knowledge independence (the formal separation and specification of knowledge types and contexts) will be essential. We repeat that the control of knowledge applications should not be removed from the purview of the various elements of expertise, but that enterprise-wide controls and guidelines must be maintained over the representation and sharing of the knowledge employed by the various applications. Obviously, for any program of this magnitude and wide-reaching impact, the support of the top officer of the department is very important. Such commitment by top management will help to provide the direction and support needed to accomplish the goal of a knowledge resource.

9.2 KNOWLEDGE-BASED SYSTEMS EXPERIMENTATION IS NEEDED

Considerable research will be required in both the managerial and technical components of a knowledge resource. In Appendix-1 we list numerous areas for research and development without attempting to assign priorities to those endeavors. An area warranting extensive investigation in the near future is that of knowledge-based systems (see Section 5). Some preliminary study and evaluation efforts are already underway in this field but a more extensive program of testing this technology against departmental knowledge applications is needed to gain valuable expertise in their use and a deeper understanding of the problems involved. The recent or imminent departure of some of the most knowledgeable and skilled civil servants emphasizes the need to organize formally and retain at least a part of their invaluable expertise. Knowledge-based systems might serve that function, but the experts must be made available to supply the

appropriate knowledge to the system in a process which can take several years to accomplish.

9.3 KNOWLEDGE INDEPENDENCE IS CRUCIAL

The issue of knowledge independence is also a critical aspect which should be considered by system analysts in designing future systems for departmental projects. In Section 8 we presented a logical system design which was intended to identify some of the basic components of a knowledge resource. The authors do not advocate that this design is the only design possible, nor even that it is the best; rather, we have included the design as a demonstration of the kind of system which a knowledge resource might be based upon. The design provides for the separation of the factual knowledge from procedural knowledge and judgmental knowledge along with the provision of some sort of translation and control mechanism to allow the different components to work together. This embodiment of knowledge independence offers the advantages of permitting data sharing (really knowledge sharing) among departmental applications, supplying a limited number of (standardized) user interfaces to the knowledge resource, and a mechanism for monitoring and understanding the flow of knowledge among the various elements of the organization from a centralized point. Another important aspect of knowledge independence is the need to recognize and separate heuristic procedural knowledge from what is thought to be algorithmic procedural knowledge. The recognition of this separation can lead to significant changes in the current methods of designing systems and writing software.

9.4 JOINT EFFORTS ARE ESSENTIAL

A department should not attempt to build a knowledge resource completely alone. No mention has been made of the expected cost of such an endeavor (indeed, a cost/benefits trade-off analysis should be performed), but there is no need for any organization to incur all of the research and development costs necessary for such a program. The Defense Advanced Research Projects Agency (DARPA), the Office of Naval Research, and the National Science Foundation (NSF) are just a few of the agencies which fund research in the information systems area. Departments should become very active in stating their requirements for and in participating in the research which these organizations sponsor. A unified departmental position on what will be needed to build and operate their knowledge resource is essential to obtaining such vital support. In addition to seeking other sources of funding, a department should also seek out and collaborate with other departments which might be interested in developing their own knowledge resources. Indeed, the development of knowledge within a given department is of little worth without the ability to communicate their knowledge to others. Departments can supply valuable knowledge to assist each other in their tasks, but a coordinated methodology will be required. Experiences with the present computer networks can attest to the utility and difficulty of such knowledge sharing among agencies. We believe that the ideas outlined in this paper can serve as a partial basis for that

methodology.

9.5 A KNOWLEDGE RESOURCE METHODOLOGY IS NEEDED

The final recommendation is that Federal departments work to develop a methodology for developing knowledge resources. We have made some recommendations in this paper concerning general specifications for system design. Much more specific thinking is necessary before the knowledge resource philosophy can be effectively institutionalized in the major Federal departments. Indeed, more facts and better arguments will be needed before top-level departmental management can be expected to commit to implementing such an approach. We recommend that task forces be appointed to study and expand the definition of a knowledge resource philosophy. The task forces should identify the short-range and long-range consequences of embracing the philosophy which they have refined. Finally, plans must be drawn up which will identify the actions which will need to be taken in order to implement knowledge resources for each of the various departments. Such a plan might contain (among others) recommendations to

* Identify existing problems where having a knowledge resource could offer improved solutions.

* Establish and nurture appropriate links between Federal departments in order to begin knowledge sharing between departments.

* Expand efforts in the test and evaluation of knowledge-based system technology.

* Establish Knowledge Resource Technology Transfer Laboratories.

* Develop a methodology for performing systems analysis under a knowledge resource philosophy.

* Develop and experiment with the logical system design concepts (new user interfaces, new knowledge managers, canonical knowledge representation and translation, and knowledge independence).

* Identify and use joint funding possibilities for knowledge resource development efforts.

REFERENCES

1. Monthly Labor Review, U.S. Bureau of Labor Statistics, June 1976.

2. Martin, James, Computer Data-Base Organization, Prentice Hall, Englewood Cliffs, N.J., 1975.

3. Auerbach Reporter, "Data Base Management: Where Are We? Where Are We Heading?" January 1975.

4. Everest, Gordon C., "Database Management Systems Tutorial," Fifth Annual Midwest AIDS Conference Proceedings, Vol. 1, Minneapolis, Minn.

5. Bontempo, Charles J., "Data Resource Management," Data Management, February 1973.

6. Edwards, N. P. and H. Tellier, "A Look at Characterizing the Design of Information Systems," Proceedings of the ACM, 1974.

7. Lyon, John K., The Database Administrator, John Wiley & Sons, New York, N.Y., 1976.

8. Ware, Willis H., "Computers, Personal Privacy and Human Choice," NTIS AD A005 315, The Rand Corporation, Santa Monica, California, December 1973.

9. Privacy Act of 1974, Public Law 93-579, U.S. Congress.

10. Turn, R. and W. Ware, "Privacy and Security in Computer Systems," NTIS AD A016 493, The Rand Corporation, Santa Monica, California, January 1975.

11. "A Symposium - Knowledge Management," Public Administration Review, November/December 1975.

12. Berry, J. F., and C. M. Cook, "Managing Knowledge as a Corporate Resource," NTIS No. AD-A029891, July 1976.

13. Simon, Herbert, The New Science of Management Decision, Harper & Brothers, New York, 1960.

14. Pople, Harry E., "On Mechanization of Abductive Logic," Proceedings of the Third International Conference on Artificial Intelligence, 1973.

15. "Markets for Data-Base Services," Frost and Sullivan Inc., New York, July 1973.

16. Myers, G. J., Reliable Software Through Composite Design, Petrocelli/Charter, New York, 1975.

17. Yourdon, E. and L. L. Constantine, Structured Design, Yourdon Inc., New York, 1976.

18. Bobrow, D. G., and A. Collins (eds.), Representation and Understanding - Studies in Cognitive Science, Academic Press, New York, 1975.

19. Wegbreit, B., "The ECL Programming System," AEIPS Conference Proceedings, FJCC, Vol. 39, 1971.

20. Martin, V. A., and R. H. Fateman, "The MACSYMA System," Proceedings of ACM Second Symposium on Symbolic and Algebraic Manipulation, S. R. Petrick, ed., Los Angeles, California, (23-25 March 1971).

21. Lesser, R. L., "Organization of the HEARSAY-II Speech Understanding System," IEEE International Joint Conference on Speech Recognition, April 1974.

22. Feigenbaum, E. A., "On Generality and Problem Solving - a case study involving the DENDRAL program," in Machine Intelligence 6, ed. by Meltzer and Michie, Edinburgh University Press, 1971.

23. Shortliffe, E. H., "MYCIN: A Rule-based Computer Program for Advising Physicians Regarding Antimicrobial Therapy Selection," STAN-CS-74-465, Computer Science Department, Stanford University, October 1974.

24. Feigenbaum, E. A., "HASP Final Report," Systems Control,Inc. ARPA Contract No. N66314-74-C01235, (special access), September 1975.

25. Anderson, R. H., and J. J. Gillogly, "Rand Intelligent Terminal Agent (RITA): Design Philosophy," R-1809-ARPA, The Rand Corporation, Santa Monica, California, January 1976.

26. Pople, H., et al., "DIALOG, A Model of Diagnostic Logic for Internal Medicine," Proceedings of the Fourth International Joint Conference on Artificial Intelligence, Tbilisi, USSR, September 1975.

27. Yakimovsky, Y., "Nondeterministic Data Base for Computerized Visual Perception," NASA CR 146578, February 1976.

28. Buchanan, B. B., et al., "Heuristic Theory Formation: Data Interpretation and Rule Formation," in Machine Intelligence 7, ed. by B. Meltzer and D. Michie, Edinburgh University Press, 1972.

29. Martin, James, Design of Man-Computer Dialogues, Prentice Hall, Englewood Cliffs, N.J., 1973.

30. Interim Report, Study Group on Data Base Management Systems, ANSI/X3/SPARC, document no. 7514TS01, February 1975.

31. Shoshani, Arie, "Data Sharing in Computer Networks," Proceedings of 1972 Western Electronic Show and Convention (WESTCON), September 1972.

32. Canaday, R. H., et al., "A Back-end Computer for Data Base Management," Communications of ACM, Vol. 17, No. 10, October 1974.

33. Cullinane, John, et al., "Commercial Data Management Processor Study," Cullinane Corporation, Wellesley, Mass., December 1975.

34. Maryanski, F., et al., "Usability and Feasibility of Back-end Mini-computers," USACSC Grant No. DAHC04-75-G-0137, U.S. Army Computer Systems Command, Ft. Belvoir, Va., June 1975.

35. Ritchie, D. M., and K. Thompson, "The UNIX Time-Sharing System," Communications of ACM, Vol. 17, No. 7, July 1974.

36. Anthony, Robert, Planning Control System. A Framework for Analysis, Harvard University Press, 1965.

37. Head, Robert V., "The Elusive MIS," Datamation, September 1970.

38. Senko, M. E., et al., "Data Structures and Accessing in Data Base Systems," IBM Systems Journal, Vol. 12, No. 1, 1973.

39. Childs, David L., "Description of a Set Theoretic Data Structure," AEIPS Conference Proceedings, FJCC, 1968.

40. Chen, Peter Pin-Shan, "The Entity-Relationship Model - Toward a Unified View of Data," ACM Transactions on Database Systems, Vol. 1, No. 1, March 1976.

ADDITIONAL BIBLIOGRAPHY

Bachman, C. W., "The Programmer as Navigator," Communications of ACM, Vol. 16, No. 11, November 1973.

Bachman, C. W., "Trends in Database Management," Proceedings of the British Computer Society Conference, London, October 1974.

Burger, John F., et al., "Semantic-Based Parsing and a Natural-Language Interface for Interactive Data Management," 13th Annual Conference of the Association for Computational Linguistics, Boston, Mass., October 1975.

Codd, E. F., "A Relational Model of Data for Large Shared Data Banks," Communications of ACM, Vol. 13, No. 6, June 1970.

Codd, E. F., "Seven Steps to Rendezvous with the Casual User," in Data Base Management, North Holland Publishing Co., 1974.

CODASYL Data Description Language Committee, Journal of Development, December 1975.

CODASYL End User Facility Task Group, Progress Report, June 1975.

Date, C. J., An Introduction to Database Systems, Addison-Wesley Publishing Co., Reading, Mass., 1975.

Davis, R., and J. King, "An Overview of Production Systems," STAN-CS-75-524, Computer Science Department, Stanford University, October 1975.

Katzan, Harry Jr., Computer Data Management and Data Base Technology, Van Nostrand Reinhold Co., New York, 1975.

Kent, W., "Describing Information," TR 03.012, IBM General Products Division, Palo Alto, California, May 1975.

Langefors, B., and B. Sundgren, Information Systems Architecture, Petrocelli/Charter, New York, 1975.

Nilsson, N. J., "Artificial Intelligence," Information Processing 74: Proceedings of IEIP Congress 74, Stockholm, Sweden, August 1974.

Palmer, Ian, Data Base Systems: A Practical Reference, Q.E.D. Information Sciences, Inc., Wellesley, Mass., June 1975.

Sundgren, Bo, Theory of Data Bases, Petrocelli/Charter, New York, 1975.

Tenenbaum, J. M., et al., "Research in Interactive Scene Analysis," NASA CR 146667, January 1976.

Winograd, T., "Five Lectures on Artificial Intelligence," NTIS No. AD-A000085, September 1974.

APPENDIX-1

PROBLEMS IN IMPLEMENTING A KNOWLEDGE RESOURCE

A-1.1 INTRODUCTION

The recognition of knowledge as a corporate resource of a department will, in most instances, be a long drawn-out process. There will be many problems to be confronted along the way; some can be foreseen, some cannot. There will be political problems, economic problems, technical problems, and managerial or structural problems. Some degree of structural reorganization may be required. A new philosophy of organizational ownership and responsibility for knowledge will have to be established with a strong sense of direction and support from top-level management. Only top-level management has the power to deal with the political, economic, and structural problems head-on. Only after these problems have been solved can a unified corporate approach to a knowledge resource succeed. Even with their solution, however, there will remain many technical problems in acquiring, representing, and transmitting knowledge. Research in these vital areas is needed now.

In this Appendix we address some of the many issues which must be faced by an organization in attempting to develop a corporate policy which establishes a knowledge resource. The list of issues, though extensive, is by no means exhaustive. It does, however, point out some of the more crucial problem areas and deficiencies of current technology in this field and indicates some topics where more research is necessary. The list is organized around a blend of the Structural and Functional Views of Section 6 of the paper with problem areas being made explicit through the Physical View.

A-1.2 CONCEPTUAL MODEL

There should be someone such as an Enterprise Administrator (EA) with overall responsibility for implementing and maintaining the knowledge resource. An EA should be concerned with the proper working of each of the functional data management areas (file systems, file management systems, data base management systems, and knowledge base management systems) and with maintaining the proper flow of data, information, and knowledge between the functional units. An EA should not be as concerned with the technical operation of each individual data manager as with the overall performance of the department with respect to its knowledge resource. The purview of the EA covers not only current day-to-day informational response, but long-term knowledge production as well. An EA, if the function is created

at all, should be a high level official who is to be responsible for the knowledge resource and as such he or she should report directly to top management. In this fashion the EA is given equal standing with the other managers who are responsible for managing other basic departmental functions.

An Enterprise Administrator will undoubtedly require the assistance of a staff in addition to needing a number of tools including a powerful knowledge resource facility to keep track of the many data elements, file structures, data bases, and flows which comprise the knowledge of the Agency. It is through the office of the EA that top management makes its requests for high-level information about the functioning of the knowledge resource of the Agency. It is the EA who constantly watches for knowledge and information bottlenecks and problem areas and who is responsible for fixing them before they become critical.

A-1.2.1 Knowledge Resource

Before embarking on a knowledge resource policy, a department must have a working definition of the meaning and implications of the knowledge-as-a-resource philosophy as it applies specifically to their department. Estimates of the political, economic, technical, and structural impact must be presented to top management if they are to avoid unfavorable surprises later on. Top management, of course, will not agree to declare knowledge to be a corporate resource without an appreciation for the significance of such a declaration. The significance of where the functions to be performed are placed will not be lost on the other levels of the organization, and this placement will greatly affect how the people perceive and deal with this new function. Assigning the newly identified functions to elements steeped in one of the three contexts (for example, the computer department) will have the effect of declaring, at least structurally, that knowledge really is not viewed as a basic resource of equal importance to all divisions of the department.

Rarely does a department act in a complete vacuum. Quite often it interacts with other enterprises which may direct its activities or function as a customer for its products. There will be varying amounts of knowledge exchanged between the department and these other organizations. Someone needs to define the degree of interaction and the relationship of the department to other organizations including the volume, direction, and importance of various data and information flows. Similarly, someone should develop a model of a macro-view of the department itself and the relationships among the various subdivisions within it. This model should include tasking requirements, data and information flows, and the existing managerial structure of the organization. In addition, the organization will need a model of its knowledge resource and of the processes by which the employees acquire and maintain their knowledge. These models not only will serve as a representation of goals and current status, but they can also provide the organization with a means for testing the impact of various hypothetical changes in

requirements or organizational structure. Current modeling techniques may be adequate for this process, but it is doubtful that any such models have actually been constructed. Experimentation in the design and use of these models is needed to uncover unexpected problems and applications.

A-1.2.2 Policy

The department will need to establish policies for the organization in data management functional areas as well as policies for the human/machine and machine/machine interfaces which might affect a knowledge resource. Selection guidelines will need to be established for data management software. Evaluation procedures, sample data bases, benchmark tests, and checks on the consistency of a proposed system with the overall data management plan of the organization will all need to be developed. The policy should offer direction for data management research in addition to operational activities. These policies will need to be coordinated with other organizations outside of the department as appropriate to insure compatability in inter-organizational transactions which must adhere to applicable standards policies. Outside organizations also frequently set the information systems standards policy which the department must adhere to, and the organization must keep abreast of such policies and, wherever possible, influence them before they are issued. Finally, it is suggested that environmental impact studies be required from those projects which propose data management systems contrary to established guidelines. These studies should address such topics as the anticipated need for knowledge sharing with other systems and plans for future migration to next-generation equipment. They should also include studies on the costs which the rest of the department might incur in the future due to the inclusion of the non-standard system. The production of useful guidelines for the preparation of meaningful impact studies is an area needing further research.

A-1.2.3 Standards

Standards is an area which few people like, but which most people feel is necessary in a resource-sharing environment. It is a necessary task to establish a policy regarding all levels of standardization as it impacts the knowledge resource in the enterprise. Someone must determine effective methods for enforcing standards policies, establish a policy for resolving conflicts between multiple standards as imposed by external organizations, and develop techniques for automatic translation between standards at all levels. In situations where there are different kinds of hardware and software which are intended to share data and knowledge, a great number of areas are potential targets for standardization. This may include techniques and procedures as well as hardware and software.

It is recognized that all of the costs of standardization must not be transferred to the end users of the data bases by requiring them all to view the data in exactly the same way.

Since some degree of standards-translation will be needed in any case, additional research is necessary to determine how much of the standards burden can be shouldered by this type of software, thus freeing users from continuous and onerous training in standards. The view of data/knowledge which one employs is a function of one's position within the departmental structure. It is also a function of the context in which that data/knowledge is used. Multiple levels of standards (which encompass a hierarchy of detail) are necessary to provide the proper perspective for different classes of users. It is unwise and undesirable to force the clerk in the field to maintain a view of the data identical to that held by the Secretary. The classical view of standards holds that this conformity is the main goal of standardization. On the contrary, the goal of standardization should be to facilitate communications among different divisions of the department by agreeing on standardized concepts, not standardized names. The terms in the daily vocabulary of the clerks are user-specific, natural, convenient, and necessarily quite detailed. On the other hand, the vocabulary of the Secretary is more general, dealing with summary information at a high level. Neither group can function with the other's vocabulary, nor should they have to, as long as the concepts of the knowledge passed between them are clear. This is the purpose of the single conceptual-schema / multiple external-schema approach outlined in Section 6.2. Of central importance is the ability to define and maintain the context of reference to particular data items (or data bases). Localized vocabulary should be adequate for local context. Only when access is required of more global data must the terminology differences be resolved. Extensive thinking and experimentation needs to be done in this area before an effective implementation can be obtained.

A-1.2.4 Training

Considerable training will be required for an Enterprise Administrator and staff, and they, in turn, will need to educate others in the policies which must be followed. The organization will require advanced knowledge in the following areas: conceptual schema design; organizational dictionary and directory development and use (see Section A-1.2.6 below); existing policy, standards, and organizational structure; departmental goals, applications, and need for knowledge sharing (both within and outside the department); data base administration; the state-of-the-art in data management; long-range planning and forecasting; and general management techniques. Many of these subject areas are highly dynamic and the organization will constantly need training to keep up-to-date in these fields. Someone in the organization must have responsibility for training the rest of the department in these areas: the philosophy of treating data and knowledge as a corporate resource; policies (see 2 above) which have been established; and the use of the conceptual schema designed for the department

A-1.2.5 Knowledge Resource Presentation

Effective management of the knowledge resource of a department may require a centralized information system which is separate and distinct from the other information systems of the enterprise and which can be used by (1) some executive staff officers to monitor the inter-system flow of information within the department and by (2) top management to answer high-level requests for information concerning the functioning of the organization in general (e.g., "where do we have information on ...?" or "what information do we have that would help us assess the impact of closing the office in ...?"). A centralized facility called the Knowledge Resource Center (KRC), which is dedicated to the support of the department as a whole appears necessary. The KRC should be a dedicated computer system with a data base containing summary information about the other data bases of the department. This data base can be expected to be driven automatically by receiving its input from the other data bases in the organization and, hence, it should be accurate and up-to-date. It should provide special user interfaces which are suitable for handling the kinds of questions which top managment asks. The KRC concept appears to be capable of supporting what Robert Anthony termed the "strategic planning" function of the enterprise [36]. This function is also recognized as the top layer of an integrated Management Information System (MIS). MIS's, to date, have been marked by a notable lack of success [37]. Perhaps one of the major reasons for this state of affairs has been the failure of implementors to recognize the need for a centralized facility such as the Knowledge Resource Center to support the MIS and to support the specific knowledge needs of top and middle management.

It must be recognized that there will be many problems to be overcome in maintaining the KRC. In particular, there will be problems in connecting the KRC with all of the schemas in the various data bases of the department. Such automatic connections are necessary, however, if the costs for the KRC are to be held within acceptable bounds. Once the data resides in the KRC there will be problems with developing techniques for automatically summarizing information, and discovering effective means for presenting this information (e.g., textual summaries, charts, graphs, voice, large graphic displays, static slides, etc.). A navigational capability will be needed to portray the structure of the entire knowledge resource and to assist in the process of determining where particular pieces of information relevant to a request may be found.

A-1.2.6 Data Dictionary / Data Directory

An integral part of the Knowledge Resource Center (see A-1.2.5 above) and something essential to effective data and knowledge sharing in a computer network is a data dictionary / data directory system which serves as a central point of definition and location for all the data items which exist in data bases throughout the department. This is also the focal point

for data standardization. It is here that the various user views and terminology are reduced to a consistent, canonical enterprise view and standard vocabulary. Each data management system may maintain its own dictionary/directory, but for the network a centralized facility must be established (1) to coordinate the individual systems, (2) to provide for semantic validation and consistency-checking of the various external and internal schemas against the conceptual schema and the conceptual schema against the real world, (3) to provide a model for data location in a distributed data base environment, and (4) to permit synonym resolution among the external schemas in an attempt to relieve the end-user from having to employ standard terminology. An automated information system will be necessary for manipulating and accessing the dictionary/directory and the question of allowing automatic additions and deletions will have to be investigated. If the dictionary/directory is maintained as an overhead superfluous function instead of being automated and fed directly from the many data bases of the Agency, it will rapidly grow out-of-date and become useless. Alternatively, non-automation may require a prohibitive expenditure of human resources to keep it up-to-date. Little is known today about how to maintain automatically a dictionary/directory of dynamic data bases, and research is badly needed to define this critical area.

A-1.2.7 Security

In the area of security, the organization must recognize the vital nature of the conceptual schema (in essence, the roadmap to all the knowledge of the department) and must take extraordinary measures to preserve its integrity and privacy. These considerations are also part of the rationale for having an independent knowledge resource center. The conceptual schema contains not only data in the traditional sense but a heavy proportion of metadata (or data about this data). To date, little research has been done concerning the security requirements for metadata. What is needed is a secure information system, which implies a secure operating system, which implies a secure computer. Lacking such an environment, the organization must perform a vulnerability assessment and develop methods and procedures for controlling access to the conceptual schema while helping to establish policies on the dissemination of knowledge from the knowledge resource.

A-1.3 INTERNAL MODEL

The role of Data Base Administrators (DBA) undoubtedly will be filled by many people. In a physical environment, where there are several diverse processing centers under the control of a department, there may be a DBA for each of these centers; or there may be a DBA in charge of each of the data management functional areas. The organization may wish to organize its DBA's under some hierarchical structure in which there is a highest authority DBA to whom all the others must eventually report. This person may also be the Enterprise Administrator, but more likely he or she will be a key staff assistant to the EA. It

must remain clear that the domain of the DBA is strictly technical. The DBA is responsible for the performance of the individual functional components as well as the technical details of inter-system communication and knowledge sharing. It is up to the DBA staff to keep track of advances in technology which can be used by the enterprise to enhance its data management activity.

A-1.3.1 Hardware

The chief technical advisor on the hardware aspects of a knowledge resource will be the Data Base Administrator who must track the state-of-the-art in hardware development for data management. Several existing activities warrant close monitoring and investigation by the DBA and the staff. These areas include: back-end processors for data management functions, centralized data management machines for a network, more intelligent disk controllers, associative processors, arrays of microprocessors for parallel specialized searching, microprogramming of data management functions, mass memory technology, a methodology of storage hierarchy management, distribution of functions among tightly linked front-end, middle, and back-end computers, and the development of new access methods. In addition to tracking current activity, the DBA should perform and maintain a state-of-the-art trend analysis and cost prediction analysis for the hardware components which are crucial in data management. The analysis should span the next 5-10 years and should include a methodology for performing hardware cost/benefit analyses. As the size and the investment in the knowledge resource grows, decisions about migration to new systems become increasingly more critical in terms of dollar cost and the impact on the department's operation. The DBA will play a key role in determining the impact, feasibility, cost, and appropriate timing of such moves.

A-1.3.2 Performance

Performance is probably the one area where data base administrators receive the most complaints from users and managers alike. There is a great need for tuning tools for hardware, software, schema design, and query processing. In addition, models of significant performance variables are also needed to allow hypothetical tuning without affecting the actual running system. These models and tools must be available for an entire network and its unique problems as well as for the individual systems within the network. Furthermore, tools are needed to measure actual data base usage and compare it against original design specifications to determine when re-structuring is warranted. A dynamic re-structuring capability in individual systems can also be a significant performance factor. Finally, there is the need to develop an evaluation procedure for determining the suitability of various access methods for different applications.

A-1.3.3 Security

In the security area there is great need for improved security techniques in all layers of hardware and software. The DBA should follow and support activities in data encipherment, user authentication, development of secure operating systems, construction of automated security aids (such as monitors, auditors, alarms, etc.), evaluation of penetration techniques, construction of a security-filter interface between the data management software and the applications or users, and finally, the development of a knowledge-based front-end computer to perform security monitoring and to make decisions and draw inferences about individual accesses to the information resource. At the very least the DBA must be responsible for insuring that the data management systems offer no additional security holes or risks to already weak operating systems.

A-1.3.4 Integrity

Several areas in the category of integrity need to be enhanced. Again, preserving the integrity of the data base is the responsibility of the DBA. Work should be done to devise a workable and agreed-upon technique for locking and for resolving deadlocks (this is especially important in a computer network). Rapid recovery from failure is also a very real issue with large operational data bases. Adequate audit-trail capabilities for back-up, integrity-checking routines for maintenance, and restoration tools for recovery all need to be developed. Finally, a technique for automatically checking the semantic consistency of data would be useful (e.g., field A must be less than 10000 if field B is between 76 and 92).

A-1.3.5 Input

The Data Base Administrator is responsible for the physical mapping of the logical data base to the physical storage devices. Tools are needed to improve the efficiency of this process and to cut down on trial and error data base design. A methodology is needed for determining in advance the expected size of a data base as well as performance characteristics to be expected. Models are necessary to allow the simulation of the effects of certain parameter changes on the performance of the system. In the area of inputting the logical structure of the data base, schema navigation tools are needed to assist the DBA in perusing and altering existing schemas. Automated procedures for migrating existing data bases to new hardware or software are essential. Techniques are needed for automatically generating schemas from diagrams or sample programs. Also, a method for keeping track of multiple versions of a schema would assist in maintaining a data base whose structure changes dynamically. Finally, there needs to be developed a procedure for determining if the internal schema as designed meets the users' or applications administrators' requirements.

A-1.3.6 Distributed Data Bases

In a multi-computer environment it may be desired to distribute data bases across several machines. The DBA's involved are jointly responsible for maintaining such an environment. They will need to support research to develop techniques for easing data migration or roll-over of applications from one system to another; to devise an effective scheme for keeping copies of a data base in synchrony; to develop ways of dynamically allocating network resources (e.g., storage, communications facilities, data management capabilities, etc.); to develop and employ subnetwork models within a computer network; and to devise techniques for solving problems which take on new significance when distributed data bases are involved (e.g., update, integrity, access limitation, locking, etc.). In some instances it may be desirable to divide a large data base into several discrete logical subsets in order to permit parallel accessing (i.e., assigning parallel subtasks to individual systems in the network). In this regard, work needs to be done to provide local and global views of a data base to enhance performance. Similarly, the division of a data base into discrete physical subsets can be used to permit physical parallel access such as with an array of microprocessors.

A-1.4 EXTERNAL MODEL

The external model incorporates both end-users and Applications Administrators (AA). In the model, these individuals are envisioned as being under the administrative control of the various divisions of the applications areas. They may work with, but do not report to, the Data Base Administrator or the Enterprise Administrator. They are applications-oriented with specific tasks to be accomplished for which the data resource is only a tool. The job of the AA is to insure that the end-users can obtain the knowledge which they need to perform their tasks, and that they can get it fairly easily. In this position the AA is largely concerned with the user interfaces: their variety, appropriateness, and power. He is also concerned with the production of knowledge by the sub-groups of the department from the knowledge resource. In order to support this position, a large number of tools and improvements to existing systems need to be developed.

A-1.4.1 Data Base Access

There are many possible user interfaces for accessing data (see Section 6.1). Each differs in the amount and kind of knowledge or skills required for the user to become proficient in their use, in the degree of difficulty and flexibility in making the request, in the amount of processing required of the machine, in the form of presentation of the response, and in the information bandwidth of the interface. These interfaces embody a spectrum of capabilities, and research is needed in each area to improve the capabilities as necessary. In some instances new technology will be required; in others, only a new application of old technology is needed. In any case, research needs to be done

on how to allow the user to make an easy transition from one interface to another.

A-1.4.2 Data Presentation

Like the Enterprise Administrator, the end-users need to increase the effectiveness of data presentation. While their data is different, the methods of presentation are similar. Technology such as superimposing images over pictures (e.g., slides or television) should be investigated to reduce the volume of data generally associated with computer-generated graphics. Human factors concerning the relative ease of interpretation of various forms of presentation such as graphs, charts, tables, or text need to be studied. An automatic exception-reporting capability in which an alerter is triggered when certain user-specified conditions occur in a data base can have great implications. Finally, methods of summarizing data from numbers, graphs, or text and presenting the summaries in the most intelligible form need to be investigated. Consideration of the information bandwidth capability of each presentation form will take on increasing importance as users tend more and more to move away from bulk computer print-outs.

A-1.4.3 Navigation

End-users and applications administrators have need for a navigational facility which will allow them to browse through a data base with an unknown schema. The data base must be capable of instructing the user as to its structure and use. Techniques are needed for determining the "optimal" path to a data item (shortest, fastest, informationally richest). Also needed are methods for generating automatically the necessary access code. Pre-retrieval query analysis (complexity or cost estimation) and query simulation can help conserve machine as well as human resources. In general, the techniques of computer-aided instruction should be applied to the task of informing the user how best to use the resources available.

A-1.4.4 Data Validation

Initial validation of input data must occur before the data is entered into the data base. Methods of automatic error detection and correction (perhaps employing semantics and context) which filter data before it is entered into the data base need to be developed. Once the data has been entered, however, further validation should be performed as an integrity check. Methods to do this are badly needed. A scheme also needs to be developed to validate derived data (or to validate the algorithms used in deriving the data). Validation techniques are needed for testing the consistency of the conceptual, internal, and external schemas. Finally, some method of associating a validity value with each data element (e.g., 0-100%) and with data bases in general needs to be developed so that meaningful validities can be assigned to information derived from multiple sources. In addition to the numerical values assigned to the validities it is

often desirable (and sometimes essential) to preserve a reference to the individual responsible for assigning the current validity values.

On output, certain probabilistic filters are needed to filter out data having a validity below a given threshold. The capability to check context integrity before releasing information is also desirable as is the ability to validate queries before they are executed.

A-1.4.5 Data Entry

Efficient techniques for handling the entry of large volumes of data are sorely needed. Rapid file management techniques can be used to capture large volumes of "live" data and to hold it for later processing, but more effective methods for passing this data on to subsequent processes are needed. Currently, arbitrary data entry user interfaces and data managers are not designed to be transportable and are, in fact, bound in specific subsets. The result is that there is a multiplicity of data entry packages, all serving essentially the same function. This is further complicated by the fact that once the data is stored via one data manager it is not easily transferred to some other data management system for further processing. Some coordination of these efforts is required in order to effect data sharing among all levels of the Functional View. From the end-user's point of view, standard data entry protocols for all processes in the enterprise would reduce training requirements and increase human efficiency. Effective knowledge representation can also assist in this process by allowing better first-line editing.

A-1.4.6 Correlation and Inferencing

The end-users want to be able to correlate data and knowledge from a variety of places in the knowledge resource. The DBA's have been assigned the task of providing the capability to access different data bases, but additional tools are needed to assist the user in drawing inferences and conclusions from multiple data sources, i.e., in abstracting knowledge. Tools are needed to model the real world, to develop and test hypotheses about the real world based on the data available, and to project the implications of a hypothesis about the real world through some simulation mechanism. Current work in Question-Answering systems and automatic inferencing needs improvement in the following areas: question presentation, interactive question enhancement, the application of probabilistic rules, fuzzy logic, basic inferencing techniques, answer presentation, proof demonstration, and inductive inferencing of rules from sample data (with applications to trend analysis). One further very interesting problem is the use of time-dependencies on queries in order to keep straight the accession of archival data bases.

In this Appendix we have tried to identify many of the technical problems which an organization can expect to face in establishing a a knowledge resource. Several research topics were

listed in conjunction with the problem areas associated with each of the three models of information. To be complete, a knowledge resource research program must at least consider these areas and determine where solutions can be expected to appear on their own and where the department must actively support research efforts to increase the chances of their success in a relevant timespan.

APPENDIX-2

UNIX PROTOTYPE

A-2.1. INTRODUCTION

Some preliminary work has begun in various government organizations in order to test the feasibility of the logical system design described in Section 8. The work is being done on the UNIX operating system [35] with Digital Equipment Corporation PDP-11 computers. It should be emphasized at the outset that the combination of UNIX and PDP 11's was selected only as a prototype for initial testing of the logical system design. No conclusions have been reached as yet concerning the practicality of employing UNIX in a production-model implementation; indeed, alternative hardware configurations such as multiple PDP-11's or some larger-scale computers must be considered. Hundreds of UNIX installations presently exist all over the country. They are found in universities, government departments, and in private organizations. In addition, UNIX has been selected to form the basis of DARPA's Intelligent Terminal program, and considerable technological fallout is expected from that effort.

A-2.2 FACTUAL KNOWLEDGE SUBSYSTEM

A relational data base management system called INGRES is available from the University of California at Berkeley and it has been installed on several UNIX systems. The product is experimental, but one which is recognized as a good implementation of relational principles. Test and evaluation procedures have begun in several organizations in order to determine the efficiency and applicability of INGRES for some live situations.

ELS Systems Engineering, Inc. of Cleveland, Ohio offers a file management system called Product 3 to the UNIX environment. The software is available for purchase.

UNIX has its own hierarchically structured file system which is recognized as a remarkable implementation among minicomputer operating systems.

A-2.3 JUDGMENT SUPPORT SUBSYSTEM

The System Development Corporation of Santa Monica, California has begun to implement a generalized natural language subset interface to INGRES (see Section A-2.2). This work is expected to be completed in 2nd Quarter FY-78.

Research efforts on a navigational interface have begun to report some of their results in the literature. An experimental

tool for navigating CODASYL schemas has been developed which displays a picture of the structure of any CODASYL data base by using computer-generated graphics.

The specifications for a forms interface have been compiled by the CODASYL End User Facilities Committee, but no implementations of a generalized forms interface are currently available.

Query languages, relational, and programmatic interfaces are available with the three data managers listed in Section A-2.2. Some future work to arrive at a standard for each category will be necessary. Efforts which will lead to graphical, menu, dialogue, and transaction processors appear to be underway.

Finally, a superb text-editor from the Rand Corporation is available for UNIX. This editor appears to have great potential for a variety of applications.

A-2.4 PROCEDURAL KNOWLEDGE SUBSYSTEM

There is a primitive knowledge-based system available on UNIX which is called RITA (see Section 5.1.6). RITA has been developed by the Rand Corporation as part of DARPA's Intelligent Terminal program. RITA is presently installed on several government computer systems, and experiments are getting underway to test the implementation and possible applications.

A-2.5 TRANSLATION AND CONTROL SUBSYSTEM

The Translation and Control Subsystem appears to require considerably more research than the other three areas. Several canonical representations and translation mapping mechanisms are under investigation at numerous research institutions throughout the country. Martin Marietta Corporation in Denver, Colorado has proposed the use of the DIAM representation developed by Dr. Michael Senko at IBM [38]. The University of Maryland in College Park, Maryland and Set-Theoretic Information Systems Corporation in Ann Arbor, Michigan are both investigating Extended Set Theory [39] as a possible canonical representation. Finally, Dr. Peter Pin-Shan Chen at MIT [40] has proposed an Entity-Relationship theory which appears to hold great promise. None of this work is currently involved with UNIX. In fact, it may not be prudent at this time to constrain the research by forcing it to fit the UNIX mold. Rather, perhaps it might be best to encourage the development of the theory first and then later move to a UNIX prototype (as has happened already in the data management area). Similar remarks apply to the distributed data base, distributed dictionary/directory, and data location research.

Initial implementations of user interfaces are intended to have direct connections to the appropriate data managers. As the elements of the Translation and Control Subsystem are developed they will need to be phased into the total system. The user interfaces must be designed so that a clean break from the data

managers and connections to the intermediate subsystems are possible. The success of the individual user interfaces and data managers should not depend on the success of the development of the Translation and Control Subsystem since useful packages can be developed with direct connections. The success of a total distributed knowledge resource is, however, dependent upon the ultimate success of the canonical representation and mapping modules since this is to be the common form for representing information among distributed systems.

Numerous other software products are now available or are under development for UNIX. These include:

* Programmers' Workbench - a system for software development

* VOTRAX - a voice synthesizer for vocal output

* Security Kernel - to enhance system security

* Graphics - a system employing raster-scan video technology

* Network Control Program - for connection to ARPAnet

* Message Subsystem - for user communication

* Project Management Subsystem - to support project management

* Document Preparation Package - for word-processing

* LISP, APL, ECL, FORTRAN IV PLUS - programming languages

These tools, together with the knowledge resource software described in the preceding sections, hold great promise for developing a very strong user-oriented system, however, it is necessary to repeat the earlier caveat that what is being done is initial research on the components of a knowledge resource. In this environment, UNIX will form the basis for an initial prototype. It is not suggested that this work will constitute the final solution.